Computational Optimization and Applications (2020) 75:35–61
https://doi.org/10.1007/s10589-019-00140-7

Markov chain block coordinate descent

Tao Sun[1] · Yuejiao Sun[2] · Yangyang Xu[3] · Wotao Yin[2]

Received: 21 November 2018 / Published online: 22 October 2019

Abstract
The method of block coordinate gradient descent (BCD) has been a powerful method for large-scale optimization. This paper considers the BCD method that successively updates a series of blocks selected according to a Markov chain. This kind of block selection is neither i.i.d. random nor cyclic. On the other hand, it is a natural choice for some applications in distributed optimization and Markov decision process, where i.i.d. random and cyclic selections are either infeasible or very expensive. By applying mixing-time properties of a Markov chain, we prove convergence of Markov chain BCD for minimizing Lipschitz differentiable functions, which can be nonconvex. When the functions are convex and strongly convex, we establish both sublinear and linear convergence rates, respectively. We also present a method of Markov chain inertial BCD. Finally, we discuss potential applications.

Keywords Block coordinate gradient descent · Markov chain · Markov chain Monte Carlo · Markov decision process · Decentralized optimization

Mathematics Subject Classification Primary 90C26; Secondary 90C40 · 68W15

The work by Y. Sun and W. Yin was supported in part by NSF DMS-1720237 and ONR N0001417121. The work by Y. Xu was supported in part by NSF DMS-1719549 and an IBM Grant.

✉ Wotao Yin
wotaoyin@math.ucla.edu

Tao Sun
nudtsuntao@163.com

Yuejiao Sun
sunyj@math.ucla.edu

Yangyang Xu
xuy21@rpi.edu

1 National Lab for Parallel and Distributed Processing, College of Computer, National University of Defense Technology, Changsha 410073, Hunan, China

2 Department of Mathematics, University of California, Los Angeles, CA 90095, USA

3 Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA

Contents

1 Introduction

We consider the following minimization problem

$$\text{minimize} \quad f(x) \equiv f(x_1, x_2, \ldots, x_N) \tag{1}$$

where $f : \mathbb{R}^N \mapsto \mathbb{R}$ is a differentiable function (possibly nonconvex) and every $\nabla_i f$ $(i = 1, 2, \ldots, N)$ is Lipschitz with constant $L > 0$.

The block coordinate gradient descent (BCD) method is a popular approach that can take the advantage of the coordinate structure in (1). The method updates one coordinate, or a block of coordinates, at each iteration, as follows. For $k = 0, 1, \ldots$, choose $i_k \in [N] := \{1, 2, \ldots, N\}$ and compute

$$x_{i_k}^{k+1} = x_{i_k}^k - \gamma \nabla_{i_k} f(x^k),$$

where γ is a step size; for remaining $j \in [N] \backslash \{i_k\}$, we keep $x_j^{k+1} = x_j^k$.

The coordinate gradient descent method was introduced in [29]. The random selection rule (i.i.d. over the iterations) appeared in [15,24]. In the same paper [15], the method of accelerated coordinate gradient descent was proposed, and it was later analyzed in [10] for both convex and strongly convex functions. Both [10,15] select a coordinate i with probability proportional to the Lipschitz constant L_i of $g(\alpha) = \nabla_i f(x + \alpha e_i)$ over free x; the rate is optimal when L_i's are equal. An improved random sampling method with acceleration was introduced in [1], which further decreases the complexity when some L_i's are significantly smaller than the rest. This method was further generalized in [8] to an asynchronous parallel method, which obtains parallel speedup to the accelerated rate. In another line of work, [6] combines stochastic coordinate gradient descent with mirror descent stochastic approximation, where a random data mini-batch is taken to update a randomly chosen coordinate. This is improved in [31], where the presented method uses each random mini-batch to update all the coordinates in a sequential fashion. Besides stochastic selection rules, there has been work of the cyclic sampling rule. The work [30] studies its convergence under the convex and nonconvex settings, and [2] proves sublinear and linear rates in the convex setting. The constants in these rates are worse than standard gradient descent though. For a family of problems, [26] obtains improved rates to match standard gradient descent (and their results also apply to the random shuffling rule). The greedy sampling rule has also been studied in the literature but unrelated to this paper.

Let us just mention some references [11,12,16,20]. Finally, [17] explores the family of problems with the structure that enables us to update a block coordinate at a much lower cost than updating all blocks in batch.

This paper introduces the Markov-chain select rule. We call our method *Markov-chain block gradient coordinate descent* (MC-BCD). In this method, i_k is selected according to a Markov chain; hence, unlike the above methods, our choice is *neither* stochastic i.i.d. (with respect to k) *nor* deterministic. Specifically, there is an underlying strongly-connected graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ with the set of vertices $\mathscr{V} := [N]$ and set of edges $\mathscr{E} \subseteq \mathscr{V} \times \mathscr{V}$. Each node $i \in \mathscr{V}$ can compute $\nabla_i f(\cdot)$ and update x_i. We call $(i_k)_{k\geq 0}$ a walk of $\mathscr{G}$ if every $(i_k, i_{k+1}) \in \mathscr{E}$. If the walk $(i_k)_{k\geq 0}$ is deterministic and visits every node at least once in every K iterations, then $(i_k)_{k\geq 0}$ is essentially cyclic; if every i_{k+1} is chosen randomly from $\{\text{neighbors of } i_k\} \cup \{i_k\}$, then we obtain MC-BCD, which is the focus of this paper. To the best of our knowledge, MC-BCD is new.

1.1 Motivations

Generally speaking, one does not use MC-BCD to accelerate i.i.d. random or cyclic BCD but for other reasons: when we are forced to take Markov chain samples because cyclic and stochastic samples are not available; Or, although cyclic and stochastic samples are available, it is easier or cheaper to take Markov chain samples. We briefly present some examples below to illustrate those motivations. Some examples are tested numerically in Sect. 6.

1.1.1 Markov chain dual coordinate ascent (MC-DCA)

The paper [25] proposes the stochastic dual coordinate gradient ascent (SDCA) to solve

$$\text{minimize}_{w\in\mathbb{R}^d} \left\{ \frac{\lambda}{2}\|w\|^2 + \frac{1}{N}\sum_{i=1}^{N} \ell_i(w^\top a_i) \right\}, \tag{2}$$

where $\lambda > 0$ is the regularization parameter, a_i is the data vector associated with ith sample, and ℓ_i is a convex loss function. Its dual problem can be formulated as

$$\text{minimize}_{\alpha\in\mathbb{R}^N} \left\{ D(\alpha) := \frac{\lambda}{2}\|A\alpha\|^2 + \frac{1}{N}\sum_{i=1}^{N} \ell_i^*(-\alpha_i) \right\}, \tag{3}$$

where $A \in \mathbb{R}^{d\times N}$ with column $A_i := \frac{a_i}{\lambda N}$, and ℓ_i^* is the conjugate of ℓ_i. By applying stochastic BCD to (3), SDCA can reach comparable or better convergence rate than stochastic gradient descent. We employ this idea and propose MC-DCA: in the kth iteration, while $\alpha_j^{k+1} = \alpha_j^k$ if $j \in [N]\backslash\{i_k\}$,

$$\alpha_{i_k}^{k+1} = \alpha_{i_k}^k - \gamma\left(\lambda A_{i_k}^\top (A\alpha^k) - \frac{\nabla \ell_{i_k}^*(-\alpha_{i_k}^k)}{N}\right) \tag{4}$$

where $(i_k)_{k\geq 0}$ is a Markov chain.

The Markov chain must come from somewhere. Consider that the data $a_1, a_2, \ldots, a_N$ are stored in a distributed fashion over a graph. Only when the graph is complete can we efficiently sample i_k i.i.d. randomly and access a_{i_k}; only when the graph has a Hamiltonian cycle can we visit the data in a cyclic fashion without visiting any node twice in each cycle. MC-DCA works under a much weaker assumption: as long as the graph is connected. Specifically, let a token hold $(\alpha_1, \alpha_2, \ldots, \alpha_N)$ and vector $(A\alpha)$, and let the token randomly walk through the nodes in the network; each node i holds data A_i and can compute $\nabla \ell_i^*$; as the token arrives at node i, the node accesses $(A\alpha)$ and α and computes $\lambda A_i^\top (A\alpha)$ and $\nabla \ell_i^*(-\alpha_i)$, which are used to update α_i and update $(A\alpha)$.

1.1.2 Future rewards in a Markov decision process

This example is a finite-state (N states) discounted Markov decision process (DMDP) for which we can compute the transition probability from any current state i to the next state, or quickly approximate it. We can use MC-BCD to compute the expected future reward vector.

Let us describe the DMDP. Entering any state i, we receive an award r_i and then take an action according to a given policy π (a state-to-action mapping). After the action is taken, the system enters a state j, $j \in [N]$, with probability $P_{i,j}$. The transition matrix $P := [P_{i,j}]_{i,j\in[N]}$ depends on the action taken and thus depends on π. The reward discount factor is $\gamma \in (0, 1)$. Our goal is to evaluate the expected future rewards of all states $i \in [N]$ for a fixed π. This step dominates the per-step computation of the policy-update iteration [28], which iteratively updates π.

For each state $i_0 := i$, the expected future reward is given as

$$v_i := \mathbb{E}_{\{i_t\}}\Big[\sum_{t=1}^{+\infty} \gamma^t r_{i_t} \mid i_0 = i\Big],$$

where the state sequence $(i_t)_{t\geq 0}$ is a Markov chain induced by the transition matrix P and r_{i_t} is the reward received at time t. The corresponding Bellman equation is $v_i = \mathbb{E}_{i_1}\big[r_i + \gamma v_{i_1} \mid i_0 = i\big] = r_i + \gamma \sum_{j\in[N]} P_{i,j} v_j$, the matrix form of which is

$$v = r + \gamma P v, \tag{5}$$

where $v = [v_1, v_2, \ldots, v_N]^T$ and $r = [r_1, r_2, \ldots, r_N]^T$.

When N is huge, solving (5) is difficult. Often we have memory to store a few N-vectors (also, N can be reduced by dimension reduction) but not an $N \times N$-matrix. Therefore, we can store the vector $P_i = [P_{i,1}, \ldots, P_{i,N}]^T$ only temporarily in each iteration. In the case where the physical principles or the rule of game are given, such as in the Tetris game, we can compute the transition probabilities P_{i_k} explicitly. Consider another scenario where P_{i_k} can not be computed explicitly but can be approximated by Monte-Carlo simulations. The simulation of transition at just one state i_k is much cheaper than that of all states. In both scenarios, we have access to P_{i_k}. This allows us

to apply MC-BCD to solve a dual optimization problem below to compute the future reward vector v,

$$\text{minimize}_v \left\{ \frac{1}{2N} \|(\mathbb{I}_N - \gamma P)v - r\|^2 + \frac{\lambda}{2}\|v\|^2 \right\},$$

where $\lambda \geq 0$ is a fixed regularization parameter. This corresponds to setting $A := \mathbb{I}_N - \gamma P$ in (3). Note that in DMDP, one cannot transit from the current state i_k to an arbitrary $j \in [N]$. Therefore, standard cyclic and stochastic BCD is not applicable.

Running the MC-DCA iteration (4) requires the vectors $A_{i_k} = P_{i_k}$ and $A\alpha^k = \alpha^k - \gamma P\alpha^k$. We update $(P\alpha^k)$ by maintaining a sequence $(w^k)_{k\geq 0}$ as follows: initialize $\alpha^0 := 0$ (zero vector) and thus $w^0 = P\alpha^0 = 0$; in the kth iteration, we compute $w^{k+1} := w^k + P_{i_k}(\alpha^{k+1} - \alpha^k)_{i_k} = P\alpha^{k+1}$, where the equality follows since α^{k+1} and α^k only differ over their i_kth component. This update is done without accessing the full matrix P.

As we showed above, running our algorithm to compute the expected future award v only requires $O(N)$ memory. Also the algorithm iterates simultaneously while the system samples its state trajectory. Suppose each policy π can be stored in $O(N)$ memory (e.g., deterministic policy) and updating π using a computed v also needs $O(N)$ memory; then, we can run a policy-update iteration with $O(N)$ memory.

1.1.3 Risk minimization by dual coordinate ascent over a tricky distribution

Let Ξ be a statistical sample space with distribution Π, and $F(\cdot) : \mathbb{R} \to \mathbb{R}$ is a proper, closed, strongly convex function. Consider the following regularized expectation minimization problem

$$\text{minimize}_{w\in\mathbb{R}^n} \ \mathbb{E}_\xi \left(F(w^\top \xi)\right) + \frac{\lambda}{2}\|w\|^2. \tag{6}$$

Since the objective is strongly convex, its dual problem is smooth. If it is easy to sample data from Π, (6) can be solved by SDCA, which uses i.i.d. samples. When the distribution Π is difficult to sample directly but has a faster Markov Chain Monte Carlo (MCMC) sampler, we can apply MC-DCA to this problem.

1.1.4 Multi-agent resource-constrained optimization

Consider the multi-agent optimization problem of N agents [4]:

$$\text{minimize } f(x_1, x_2, \ldots, x_N) + \frac{\beta}{2}\|\max\{Ax - b, \mathbf{0}\}\|^2, \tag{7}$$

where f is the cost function, b is the resource vector, and $\max\{Ax - b, \mathbf{0}\}$ penalizes any over usage of resources. Define a graph, in which every node is an agent and every edge connects a pair of agents that either depend on one another in f or share at least one resource. In other words, the objective function (7) has a graph structure in that computing the gradient of x_i requires only the information of the adjacent agents of i.

MC-BCD becomes a decentralized algorithm: after an agent i_k updates its decision variable x_{i_k}, it broadcasts x_{i_k} to one of its neighbors, i_{k+1} and activates it to run next step. In this process, $i_0, i_1, \ldots$ form a random walk over the graph and, therefore, is a Markov chain. As long as the network is connected, a central coordinator is no more necessary. However, sampling i_k i.i.d. randomly requires a central coordinator and will consume more communication since it may communicate beyond neighbors. Also selecting i_k essentially cyclically requires a tour of the graph, which relies on the knowledge of the graph topology.

When f is differentiable with Lipschitz continuous gradient, so is the objective function. We apply MC-BCD to (7) to obtain

$$x_{i_k}^{k+1} = x_{i_k}^k - \gamma \nabla_{i_k} f(x^k) - \gamma\beta A_{i_k}^\top \max\{Ax^k - b, \mathbf{0}\},$$

where $(i_k)_{k\geq 0} \subseteq [N]$ is a Markov chain. We assume that agent i can access A_i and b_i and compute $\nabla_i f$. Similar to the example for computing expected future reward above, $v^k := Ax^k - b$ can be updated along with the iterations so no node needs the access to the full matrix A. Alternatively, we can use a central governor which receives updated x^k and v^k from agent i_k and sends the data to i_{k+1} for the next iteration.

1.1.5 Decentralized optimization

This example is taken from [32]. Again consider the empirical risk minimization problem (2). We consider solving its dual problem (3) in a network by assigning each sample a_i to a node. A parallel distributed algorithm will update for all the components, $i = 1, \ldots, N$, concurrently.

If the network has a central server, then each node sends its α_i to the central server, which forms $A\alpha = \sum_{i=1}^N A_i\alpha_i$ and then broadcasts it back to the nodes.

If the network does not have a central server, then we can form $A\alpha$ either running a decentralized gossip algorithm or calling an all-reduce communication. The former does not require the knowledge of the network topology and is an iterative method. The latter requires the topology and takes at least $O(\log N)$ rounds and at least $O(N)$ total communication, even slower when the network is sparse. An alternative approach is to create a token that holds $A\alpha$ and follows a random walk in the network. The token acts like a traveling center. When the token arrives at a node i_k, the node updates its α_{i_k} using the token's $A\alpha$, and this local update leads to a sparse change to $A\alpha$; updating $A\alpha$ requires no access to α_j for $j \neq i_k$. The method in [32] applies this idea to an ADMM formulation of the decentralized consensus problem (rather than BCD in this paper) and shows that total communication is significantly reduced.

1.2 Difficulty of the convergence proofs: biased expectation

Sampling according to a Markov chain is neither (essential) cyclic nor i.i.d. stochastic. No matter how large K is, it is still possible that a node is never visited during some $k+1, \ldots, k+K$ iterations. Unless the graph $\mathcal{G}$ is a complete graph (every node is directly connected with every other node), there are nodes i, j *without* an edge

connecting them, i.e., $(i, j) \notin \mathscr{E}$. Hence, given $i_{k-1} = i$, it is *impossible* to have $i_k = j$. So, no matter how one selects the sampling probability $p_j = \mathbb{P}(i_k = j)$ and step size γ_k, we generally do *not* have $\mathbb{E}_{i_k}(\gamma_k \nabla_{i_k} f(x^k) \mid i_{k-1} = i) = C \nabla f(x^k)$ for any constant C, where $\nabla_{i_k} f(x^k) := [0, \ldots, 0, \nabla_{i_k} f(x^k), 0, \ldots, 0]^T$. This, unfortunately, breaks down all the existing analyses of stochastic BCD since they all need a non-vanishing probability for every block $1, \ldots, N$ to be selected.

1.3 Proposed method and contributions

Given a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$, MC-BCD is written mathematically as

$$\text{sample } i_k \in \{j : (i_{k-1}, j) \in \mathscr{E}\} \sim P_{i_{k-1}, j}(k), \tag{8a}$$

$$\text{compute } x_{i_k}^{k+1} = x_{i_k}^k - \gamma \nabla_{i_k} f(x^k), \tag{8b}$$

where γ is a constant stepsize, and $P(k)$ is the transition matrix in the kth step (details given in Sect. 2), and we maintain $x_j^{k+1} = x_j^k$ for all $j \neq i_k$. The initial point x^0 can be chosen arbitrarily. The block i_0 can be chosen either deterministically or randomly. The following diagram illustrates the influential relations of x^0 and the random variable sequences $(i_k)_{k \geq 0}$ and $(x^k)_{k \geq 1}$:

$$\begin{array}{ccccccccccc} & & i_0 & \longrightarrow & i_1 & \longrightarrow & i_2 & \longrightarrow & i_3 & \longrightarrow & \ldots \\ & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \\ x^0 & \longrightarrow & x^1 & \longrightarrow & x^2 & \longrightarrow & x^3 & \longrightarrow & x^4 & \longrightarrow & \ldots \end{array}$$

To our best knowledge, (8) did not appear before and, as explained above, is not a special case of existing BCD analyses. When the Markov chain $(i_k)_{k \geq 0}$ has a finite mixing time and problem (1) has a lower bounded objective, we show that using $\gamma \in (0, 2/L)$ ensures $\mathbb{E}\|\nabla f(x^k)\| \to 0$. The concept of mixing time is reviewed in the next section. In addition, when f is convex and coercive, we show that $\mathbb{E} f(x^k) \to \min f$ at the rate of $O(1/k)$ with a hidden constant related to the mixing time. Note that running the algorithm itself requires no knowledge about the mixing time of the chain. Furthermore, when f is (restricted) strongly convex, then the rate is improved to be linear, unsurprisingly. Although we do not develop any Nesterov-kind acceleration in this paper, a heavy-ball-kind inertial MC-BCD is presented and analyzed because the additional work is quite small. When the computation $\nabla_{i_k} f(x^k)$ is noisy, as long as the noise is square summable (which is weaker than being summable), MC-BCD still converges.

1.4 Possible future work

We mention some future improvements of MC-BCD, which will require significantly more work to achieve. First, it is possible to accelerate MC-BCD using both Nesterov-kind momentum and optimizing the transition probability. Second, it is important to parallelize MC-BCD, for example, to allow multiple random walks to simultaneously

update different blocks [7,22], even in an asynchronous fashion like [13,19,27]. Third, it is interesting to develop a primal–dual type MC-BCD, which would apply to a model-free DMDP along a single trajectory. Yet another line of work applies block coordinate update to linear and nonlinear fixed-point problems [5,17,18] because it can solve optimization problems in imaging and conic programming, which are equipped with nonsmooth, nonseparable objectives, and constraints.

2 Preliminaries

2.1 Markov chain

We recall some definitions and properties of the Markov chain that we use in this paper.

Definition 1 *(finite-state (time-homogeneous) Markov chain)* A stochastic process $X_1, X_2, \ldots$ in a finite state space $[N] := \{1, 2, \ldots, N\}$ is called Markov chain with transition matrices $(P(k))_{k\geq 0}$ if, for $k \in \mathbb{N}$, $i, j \in [N]$, and $i_0, i_1, \ldots, i_{k-1} \in [N]$, we have

$$\mathbb{P}(X_{k+1} = j \mid X_0 = i_0, X_1 = i_1, \ldots, X_k = i) = \mathbb{P}(X_{k+1} = j \mid X_k = i) = P_{i,j}(k).$$

The chain is time-homogeneous if $P(k) \equiv P$ for some constant matrix P.

Let the probability distribution of X_k be denoted as the row vector $\pi^k = (\pi_1^k, \pi_2^k, \ldots, \pi_N^k)$, that is, $\mathbb{P}(X_k = j) = \pi_j^k$. Each π^k satisfies $\sum_{i=1}^N \pi_i^k = 1$. Obviously, it holds $\pi^{k+1} = \pi^k P(k)$. When the Markov chain is time-homogeneous, we have $\pi^k = \pi^{k-1} P$ and $\pi^k = \pi^{k-1} P = \cdots = \pi^0 P^k$, for $k \in \mathbb{N}$, where P^k is the kth power of P.

Definition 2 A time-homogeneous Markov chain is irreducible if, for any $i, j \in [N]$, there exists k such that $(P^k)_{i,j} > 0$. State $i \in [N]$ is said to have a period d if $P_{i,i}^k = 0$ whenever k is *not* a multiple of d and d is the greatest such integer. If $d = 1$, then we say state i is aperiodic. If every state is aperiodic, the Markov chain is said to be aperiodic.

Any time-homogeneous, irreducible, and aperiodic Markov chain has a stationary distribution $\pi^* = \lim_k \pi^k = [\pi_1^*, \pi_2^*, \ldots, \pi_N^*]$ with $\sum_{i=1}^N \pi_i^* = 1$ and $\min_i\{\pi_i^*\} > 0$, and $\pi^* = \pi^* P$. This is a sufficient but not necessary condition to have such π^*. If the Markov fails to be time-homogeneous,[1] it may still have a stationary distribution under additional assumptions.

In this paper, we make the following assumption, which always holds for time-homogeneous, irreducible, and aperiodic Markov chain and may hold for more general Markov chains.

[1] The time-homogeneous, irreducible, and aperiodic Markov chain is widely used; however, in practical problems, the Markov chain may not satisfy the time-homogeneous assumption. For example, in a mobile, if the network connectivity structure is changing all the time, then the set of the neighbors of an agent is time-varying [9].

Assumption 1 The Markov chain $(X_k)_{k\geq 0}$ has the transition matrices $(P(k))_{k\geq 0}$ and the stationary distribution π^*. Define

$$\Phi(m,n) := P(m)P(m+1)\cdots P(m+n), \quad m,n\geq 0, \qquad \Pi^* := \begin{bmatrix} \pi^* \\ \pi^* \\ \vdots \\ \pi^* \end{bmatrix} \in \mathbb{R}^{N\times N},$$

that is, every row of Π^* is π^*. For each $\epsilon > 0$, there exists $\tau_\epsilon \geq 1$ such that spectral norm

$$\|\Phi(m,n) - \Pi^*\|_2 < \epsilon, \quad \text{for all } m\geq 0, n\geq \tau_\epsilon - 1.$$

Here, τ is called a *mixing time*, which specifies how long a Markov chain evolves close to its stationary distribution. The literature has a thorough investigation of various kinds of mixing times [3]. Previous mixing time focuses on bounding the difference between π^k and the stationary distribution π^*. Our version is just easier to use in the analysis.

For a time-homogeneous, irreducible, and aperiodic Markov chain with the transition matrix P, $\Phi(m,n) = P^{n+1}$. It is easy to have τ_ϵ as $(1+\frac{3\ln N}{2\ln\frac{1}{\lambda_2(P)}})\cdot\log_{\frac{1}{\lambda_2(P)}}(\frac{1}{\epsilon})$, where $\lambda_2(P)$ denotes the second largest eigenvalue of P (positive and smaller than 1) [14]. Besides the time-homogeneous, irreducible, and aperiodic Markov chain, some other non-time-homogeneous chains can also have a geometrically-convergent $\Phi(m,n)$. An example is presented in [21].

2.2 Notation and constants

The following notation is used throughout this paper:

$$\Delta^k := x^{k+1} - x^k. \tag{9}$$

In MC-BCD iteration, only the block $\Delta^k_{i_k}$ of Δ^k is nonzero; other blocks are zero. Let $\pi^*_{\min}$ be the minimal stationary distribution, i.e.,

$$\pi^*_{\min} := \min_{1\leq i\leq N}\{\pi^*_i\}.$$

For any closed proper function f, argmin f denotes the set $\{x\in\mathbb{R}^N \mid f(x) = \min f\}$, and $\|\cdot\|$ denotes the ℓ_2 norm. Through the proofs, we use the following sigma algebra

$$\chi^k := \sigma(x^1, x^2, \ldots, x^k, i_0, i_1, \ldots, i_{k-1}).$$

Let Assumption 1 hold. In our proofs, we let τ be the $\frac{\pi^*_{\min}}{2}$-mixing time, i.e.,

$$\|\Phi(m,n) - \Pi^*\|_2 \leq \frac{\pi^*_{\min}}{2}, \quad \text{whenever } n\geq \tau - 1. \tag{10}$$

With direct calculations,

$$\frac{\pi^*_{\min}}{2} \le [\Phi(m,n)]_{i,j}, \text{ for any } i,j \in \{1,2,\ldots,N\}, n \ge \tau - 1. \tag{11}$$

If the Markov chain promises a geometric rate, then we have

$$\tau = O\left(\ln \frac{2}{\pi^*_{\min}}\right).$$

It is worth mentioning that, for a complete graph where all nodes are connected to each other, we have a Markov chain with $\tau = 1$, and our MC-BCD will reduce to random BCD [15].

3 Markov chain block coordinate gradient descent

In this section, we study the convergence properties of the MC-BCD for problem (1). The discussion covers both convex and nonconvex cases. We show that the MC-BCD can converge if the stepsize γ is taken as the same as that in traditional BCD. For convex problems, sublinear convergence rate is established, and for strongly convex cases, linear convergence is shown.

Our analysis is conducted to an inexact version of the MC-BCD, which allows error in computing partial gradients:

$$x_j^{k+1} = \begin{cases} x_j^k - \gamma\left(\nabla_j f(x^k) + \epsilon^k\right), & \text{if } j = i_k \\ x_j^k, & \text{if } j \ne i_k, \end{cases} \tag{12}$$

where i_k is sampled in the same way as in (8a), and ϵ^k denotes the error in the kth iteration. If ϵ^k vanishes, the above updates reduce to the MC-BCD in (8).

3.1 Convergence analysis

The results in this section applies to both convex and nonconvex cases, and they rely on the following assumption.

Assumption 2 The set of minimizers of function f is nonempty, and $\nabla_i f$ is Lipschitz continuous about x_i with constant $L > 0$ for each $i = 1, 2, \ldots, N$, namely,

$$\|\nabla_i f(x) - \nabla_i f(x + \alpha e_i)\| \le L\|\alpha\|, \quad \forall x \in \mathbb{R}^N, \forall \alpha \in \mathbb{R}, \tag{13}$$

where e_i denotes the ith standard basis vector in $\mathbb{R}^N$. In addition, ∇f is also Lipschitz continuous about x with constant L_r, namely,

$$\|\nabla f(x) - \nabla f(x + s)\| \le L_r\|s\|, \quad \forall x \in \mathbb{R}^N, \forall s \in \mathbb{R}^N. \tag{14}$$

We call $\kappa = \frac{L_r}{L}$ the condition number.

When (13) holds for each i, we have

$$f(x + de_i) \le f(x) + \langle \nabla_i f(x), d\rangle + \frac{L}{2}\|d\|^2. \tag{15}$$

Lemma 1 below is very standard. It bounds the square summation of Δ^k by initial objective error and iteration errors. Lemmas 2 and 3 are new; they study the bounds on $\|\nabla_{i_k} f(x^{k-\tau+1})\|^2$ because the sampling bias prevents us from directly bounding $\|\nabla_{i_k} f(x^k)\|^2$. The bounds in these three lemmas are combined in Theorem 1 to get the convergence rates of $\|\nabla f(x^k)\|$.

Lemma 1 *Under Assumption 2, let $(x^k)_{k\ge 0}$ be generated by the inexact MC-BCD (12) with any constant stepsize $0 < \gamma < \frac{2}{L}$. Then for any k,*

$$\sum_{t=0}^{k} \|\Delta^t\|^2 \le \frac{4\gamma}{2 - L\gamma} \cdot \big(f(x^0) - \min f\big) + \frac{4\gamma^2}{(2 - L\gamma)^2} \sum_{t=0}^{k} \|\epsilon^t\|^2. \tag{16}$$

Proof Recalling the definition of Δ^k in (9) and noting $x_j^{k+1} = x_j^k$ for all $j \ne i_k$, we have:

$$\langle \Delta^k, \nabla f(x^k)\rangle = \Big\langle x_{i_k}^{k+1} - x_{i_k}^k, \nabla_{i_k} f(x^k)\Big\rangle = -\frac{1}{\gamma}\|\Delta^k\|^2 + \Big\langle \epsilon^k, x_{i_k}^k - x_{i_k}^{k+1}\Big\rangle, \tag{17}$$

where we have used the update rule in (12) to obtain the second equality. By (15) and (17), it holds that

$$\begin{aligned} f(x^{k+1}) &\le f(x^k) + \langle \Delta^k, \nabla f(x^k)\rangle + \frac{L}{2}\|\Delta^k\|^2 \\ &= f(x^k) + \Big(\frac{L}{2} - \frac{1}{\gamma}\Big)\|\Delta^k\|^2 + \langle \epsilon^k, x_{i_k}^k - x_{i_k}^{k+1}\rangle \qquad (18) \\ &\overset{a)}{\le} f(x^k) + \Big(\frac{L}{4} - \frac{1}{2\gamma}\Big)\|\Delta^k\|^2 + \frac{\gamma\|\epsilon^k\|^2}{2 - L\gamma}, \qquad (19) \end{aligned}$$

where $a)$ is from the Young's inequality $\langle \epsilon^k, x_{i_k}^k - x_{i_k}^{k+1}\rangle \le \frac{\gamma}{2-L\gamma}\|\epsilon^k\|^2 + \frac{2-L\gamma}{4\gamma}\|\Delta^k\|^2$. Summing (19), rearranging terms, and noting $f(x^k) \ge \min f, \forall k$, we obtain the desired result and complete the proof. □

Also, we can bound partial gradient by the iterate change Δ^k and error term ϵ^k as follows.

Lemma 2 *Assume (14). Let $(x^k)_{k\ge 0}$ be generated by the inexact MC-BCD (12). Then for $k \ge \tau$, it holds*

$$\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 \le 2L_r^2 \cdot (\tau - 1) \cdot \sum_{d=k-\tau+1}^{k-1} \|\Delta^d\|^2 + \frac{4}{\gamma^2}\|\Delta^k\|^2 + 4\|\epsilon^k\|^2. \tag{20}$$

Proof By the update rule in (12) and the definition of Δ^k, we have $-\nabla_{i_k} f(x^k) = \frac{\Delta^k_{i_k}}{\gamma} + \epsilon^k$. Applying the triangle inequality to the above inequality yields

$$\begin{aligned}\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 \le\ & 2\|\nabla_{i_k} f(x^{k-\tau+1}) - \nabla_{i_k} f(x^k)\|^2 + 2\left\|\frac{\Delta^k_{i_k}}{\gamma} + \epsilon^k\right\|^2 \\ \le\ & 2\|\nabla_{i_k} f(x^{k-\tau+1}) - \nabla_{i_k} f(x^k)\|^2 + \frac{4}{\gamma^2}\|\Delta^k\|^2 + 4\|\epsilon^k\|^2.\end{aligned} \tag{21}$$

Note $\|\nabla_{i_k} f(x^{k-\tau+1}) - \nabla_{i_k} f(x^k)\|^2 \le \|\nabla f(x^{k-\tau+1}) - \nabla f(x^k)\|^2$. Hence, it follows from the triangle inequality and the Lipschitz continuity of ∇f in (14) that

$$\begin{aligned}\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 &\le 2\|\nabla f(x^{k-\tau+1}) - \nabla f(x^k)\|^2 + \frac{4}{\gamma^2}\|\Delta^k\|^2 + 4\|\epsilon^k\|^2 \\ &\le 2\cdot(\tau-1)\cdot\sum_{d=k-\tau+1}^{k-1}\|\nabla f(x^{d+1}) - \nabla f(x^d)\|^2 + \frac{4}{\gamma^2}\|\Delta^k\|^2 + 4\|\epsilon^k\|^2 \\ &\le 2L_r^2\cdot(\tau-1)\cdot\sum_{d=k-\tau+1}^{k-1}\|\Delta^d\|^2 + \frac{4}{\gamma^2}\|\Delta^k\|^2 + 4\|\epsilon^k\|^2,\end{aligned}$$

which gives the desired result. □

Remark 1 If $\epsilon^k = 0,\ \forall k$, then starting from (21) and by the same arguments, we can have

$$\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 \le 2L_r^2\cdot(\tau-1)\cdot\sum_{d=k-\tau+1}^{k-1}\|\Delta^d\|^2 + \frac{2}{\gamma^2}\|\Delta^k\|^2.$$

Furthermore, we can lower bound full gradient by conditional partial gradient.

Lemma 3 *Let* (10) *hold. Then it holds*

$$\mathbb{E}\big(\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 \mid \chi^{k-\tau+1}\big) \ge \frac{\pi^*_{\min}}{2}\|\nabla f(x^{k-\tau+1})\|^2. \tag{22}$$

Proof Taking conditional expectation, we have

$$\mathbb{E}\big(\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 \mid \chi^{k-\tau+1}\big) = \sum_{j=1}^{N}\|\nabla_j f(x^{k-\tau+1})\|^2\cdot\mathbb{P}(i_k = j \mid \chi^{k-\tau+1}).$$

By the Markov property, it holds $\mathbb{P}(i_k = j \mid \chi^{k-\tau+1}) = \mathbb{P}(i_k = j \mid i_{k-\tau}) = [\Phi(k-\tau, \tau-1)]_{i_{k-\tau}, j}$. Then the desired result is obtained from (11) and the fact $\sum_{i=1}^N \|\nabla_i f(\cdot)\|^2 = \|\nabla f(\cdot)\|^2$. □

Theorem 1 *Let Assumptions 1 and 2 hold and $(x^k)_{k\geq 0}$ be generated by the inexact MC-BCD (12) with any constant stepsize $0 < \gamma < \frac{2}{L}$. We have the following results:*

1. ***Square summable noise:*** *If the noise sequence satisfy $\sum_{k=0}^{\infty} \|\epsilon^k\|^2 = \mathscr{E} < +\infty$. Then,*

$$\lim_{k\to\infty} \mathbb{E}\|\nabla f(x^k)\| = 0, \tag{23}$$

and

$$\mathbb{E}\left[\min_{1\leq t\leq k} \|\nabla f(x^t)\|^2\right] \leq \frac{2}{(k+1)\pi^*_{\min}}\left[C_1(\tau)\cdot\left(f(x^0) - \min f\right) + \left(C_2(\tau)+4\right)\mathscr{E}\right]. \tag{24}$$

2. ***Non-square-summable noise:*** *If $\|\epsilon^k\|^2 \leq S$, $\forall k \geq 0$ for some positive number $S > 0$, then*

$$\mathbb{E}\left[\min_{1\leq t\leq k} \|\nabla f(x^t)\|^2\right] \leq \frac{2}{(k+1)\pi^*_{\min}} C_1(\tau)\cdot\left(f(x^0) - \min f\right) + \frac{2}{\pi^*_{\min}}\left(\frac{C_2(\tau)(k+\tau)}{k+1} + 4\right) S. \tag{25}$$

The constants used above are

$$C_1(\tau) := \frac{4\gamma}{2-L\gamma}\left(2L_r^2(\tau-1)^2 + \frac{4}{\gamma^2}\right),$$
$$C_2(\tau) := \frac{4\gamma^2}{(2-L\gamma)^2}\left(2L_r^2(\tau-1)^2 + \frac{4}{\gamma^2}\right). \tag{26}$$

Proof In the case of square summable noise, we have $\epsilon^k \to \mathbf{0}$ as $k \to \infty$. In addition, it follows from (16) that $\sum_{k=0}^{\infty} \|\Delta^k\|^2 < +\infty$ and thus $\Delta^k \to \mathbf{0}$ as $k \to \infty$. Hence, (20) implies

$$\lim_{k\to\infty} \|\nabla_{i_k} f(x^{k-\tau+1})\|^2 = 0. \tag{27}$$

Taking expectation on (27) and using the Lebesgue dominated convergence theorem, we have

$$\lim_{k\to\infty} \mathbb{E}\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 = 0.$$

Hence from (22), it follows that

$$\lim_{k\to\infty} \mathbb{E}\|\nabla f(x^k)\|^2 = \lim_{k\to\infty} \mathbb{E}\|\nabla f(x^{k-\tau+1})\|^2 \leq \frac{2}{\pi^*_{\min}} \lim_{k\to\infty} \mathbb{E}\|\nabla_{i_k} f(x^{k-\tau+1})\|^2 = 0,$$

and thus (23) holds by the Jensen's inequality $(\mathbb{E}\|\nabla f(x^k)\|)^2 \le \mathbb{E}\|\nabla f(x^k)\|^2$.

Note $\sum_{t=\tau-1}^{k}\sum_{d=t-\tau+1}^{t-1}\|\Delta^d\|^2 \le (\tau-1)\sum_{d=0}^{k-1}\|\Delta^d\|^2$ for any $k \ge \tau$. Therefore, summing both sides of (20) yields

$$\sum_{t=\tau-1}^{k}\|\nabla_{i_t} f(x^{t-\tau+1})\|^2 \le 2L_r^2(\tau-1)^2\sum_{d=0}^{k-1}\|\Delta^d\|^2 + \frac{4}{\gamma^2}\sum_{t=\tau-1}^{k}\|\Delta^t\|^2 + 4\sum_{t=\tau-1}^{k}\|\epsilon^t\|^2$$
$$\le \left(2L_r^2(\tau-1)^2 + \frac{4}{\gamma^2}\right)\sum_{t=0}^{k}\|\Delta^t\|^2 + 4\sum_{t=\tau-1}^{k}\|\epsilon^t\|^2. \tag{28}$$

The inequality in (28) together with (16) and the assumption on ϵ^k gives

$$\sum_{t=\tau-1}^{\infty}\|\nabla_{i_t} f(x^{t-\tau+1})\|^2 \le C_1(\tau)\cdot\big(f(x^0) - \min f\big) + \big(C_2(\tau)+4\big)\mathscr{E}, \tag{29}$$

where $C_1(\tau)$ and $C_2(\tau)$ are defined in (26). In addition, we have

$$(k+1)\cdot\mathbb{E}\left[\min_{0\le t\le k}\|\nabla f(x^t)\|^2\right] \le \sum_{t=0}^{k}\mathbb{E}\|\nabla f(x^t)\|^2 = \sum_{t=\tau-1}^{k+\tau-1}\mathbb{E}\|\nabla f(x^{t-\tau+1})\|^2$$
$$\le \frac{2}{\pi^*_{\min}}\sum_{t=\tau-1}^{k+\tau-1}\mathbb{E}\|\nabla_{i_t} f(x^{t-\tau+1})\|^2, \tag{30}$$

where the last inequality follows from (22). Now the result in (24) is obtained from the above inequality together with that in (29).

In the case of $\|\epsilon^k\|^2 \le S,\ \forall k \ge 0$, we have from (16) and (28) that

$$\sum_{t=\tau-1}^{k}\|\nabla_{i_t} f(x^{t-\tau+1})\|^2 \le \left(2L_r^2(\tau-1)^2 + \frac{4}{\gamma^2}\right)\left(\frac{4\gamma}{2-L\gamma}\cdot\big(f(x^0)\right.$$
$$\left. - \min f\big) + \frac{4\gamma^2(k+1)S}{(2-L\gamma)^2}\right) + 4(k-\tau+2)S.$$

In the above inequality, setting k to $k+\tau-1$ and using (30) give the result in (25). □

Although MC-BCD has sample bias, we can still use a constant stepsize. In fact, Theorem 1 indicates the stepsize can be as large as traditional BCD. The assumption on the noise sequence is weaker than the commonly used assumption $\sum_k \|\epsilon_k\| < +\infty$. When the noise sequence is non-diminishing, we have a final error that approximately matches the noise level. This is useful in an application in Sect. 5, where computing $\nabla_{i_k} f$ may involve certain sampling that becomes too expensive to require asymptotically vanishing noise.

3.2 Convergence rates for convex minimization

When f is convex, we can estimate the rates of expected objective error. We let

$$F_t := \mathbb{E} f(x^{t\cdot\tau}) - \min f \quad \text{and} \quad \overline{x} = \text{Proj}_{\text{argmin } f}(x).$$

First, we present an important technical lemma, which will be used to derive both sublinear and linear convergence results.

Lemma 4 *Let $(x^k)_{k\geq 0}$ be generated by MC-BCD (8b) with $0 < \gamma < \frac{2}{L}$. When f is convex, we have*

$$F_t^2 \leq C_\tau \cdot (F_t - F_{t+1}) \cdot \mathbb{E}\|x^{t\cdot\tau} - \overline{x^{t\cdot\tau}}\|^2, \tag{31}$$

where the constant is

$$C_\tau := \frac{\max\left\{4L_r^2 \cdot (\tau-1), \frac{4}{\gamma^2}\right\}}{\left(\frac{1}{\gamma} - \frac{L}{2}\right) \cdot \pi^*_{\min}}. \tag{32}$$

Proof Since $\epsilon^k = 0,\ \forall k$, taking expectations of both sides of (20) and using (22) yield

$$\mathbb{E}\|\nabla f(x^{k-\tau+1})\|^2 \leq \frac{\max\left\{4L_r^2 \cdot (\tau-1), \frac{4}{\gamma^2}\right\}}{\pi^*_{\min}} \cdot \sum_{d=k-\tau+1}^{k} \mathbb{E}\|\Delta^d\|^2. \tag{33}$$

For each d, we have from (18) with $\epsilon^k = 0$ that

$$\mathbb{E}\|\Delta^d\|^2 \leq \frac{\mathbb{E} f(x^d) - \mathbb{E} f(x^{d+1})}{\frac{1}{\gamma} - \frac{L}{2}}. \tag{34}$$

Substituting (34) into (33) and recalling the definition of C_τ in (32) give

$$\mathbb{E}\|\nabla f(x^{k-\tau+1})\|^2 \leq C_\tau \left[\mathbb{E} f(x^{k-\tau+1}) - \mathbb{E} f(x^{k+1})\right]. \tag{35}$$

For any integer t, letting $k = (t+1)\cdot\tau - 1$ in (35), we have

$$\mathbb{E}\|\nabla f(x^{t\cdot\tau})\|^2 \leq C_\tau \left[F_t - F_{t+1}\right]. \tag{36}$$

On the other hand, it follows from convexity of f that

$$F_t = \mathbb{E} f(x^{t\cdot\tau}) - \min f \leq \mathbb{E}\left\langle \nabla f(x^{t\cdot\tau}), x^{t\cdot\tau} - \overline{x^{t\cdot\tau}}\right\rangle. \tag{37}$$

Now square both sides of (37) and apply the Cauchy–Schwarz inequality to have

$$F_t^2 \leq \mathbb{E}\|\nabla f(x^{t\cdot\tau})\|^2 \cdot \mathbb{E}\|x^{t\cdot\tau} - \overline{x^{t\cdot\tau}}\|^2. \tag{38}$$

Substituting (36) into (38) yields (31), and we complete the proof. □

3.2.1 Sublinear convergence rate

A well-known result in convergence analysis is that a nonnegative sequence $(a_k)_{k\geq 0}$ that obeying $a_{k+1} \leq a_k$ and $a_{k+1} \leq a_k - \eta a_k^2$, for some $\eta > 0$ and all $k \geq 0$ satisfies

$$a_k \leq \frac{a_0}{a_0 \eta k + 1}. \tag{39}$$

It can be proved by observing $\frac{1}{a_{k+1}} - \frac{1}{a_k} \geq \eta$.

Theorem 2 *Under Assumptions 1 and 2, let $(x^k)_{k\geq 0}$ be generated by MC-BCD (8b) with $0 < \gamma < \frac{2}{L}$. Assume that f is convex and the level set $\mathscr{X}_0 = \{x \in \mathbb{R}^N : f(x) \leq f(x^0)\}$ is bounded with diameter $R = \max_{x,y\in\mathscr{X}_0} \|x - y\|$. Then we have*

$$\mathbb{E} f(x^k) - \min f \leq \frac{F_0 C_\tau R^2}{F_0 \lfloor \frac{k}{\tau} \rfloor + C_\tau R^2},$$

where C_τ is the constant defined in (32), and τ is the $\frac{\pi^*_{\min}}{2}$-mixing time defined in (10). □

Proof From (18) with $\epsilon^k = 0$, $\forall k$ and $0 < \gamma < \frac{2}{L}$, it follows that $f(x^k)$ is monotonically nonincreasing about k, and thus $x^k \in \mathscr{X}_0$ for all k. Therefore, $\|x^{t\cdot\tau} - \overline{x^{t\cdot\tau}}\|^2 \leq R^2$, $\forall t$. Substituting this inequality into (31) gives $F_t^2 \leq C_\tau R^2 \cdot (F_t - F_{t+1})$, or equivalently $F_{t+1} \leq F_t - \frac{F_t^2}{C_\tau R^2}$. From (39) we obtain

$$F_t \leq \frac{F_0}{\frac{F_0 t}{C_\tau R^2} + 1}, \ \forall t \geq 0.$$

Since $f(x^k)$ is nonincreasing about k, it follws that

$$\mathbb{E} f(x^k) - \min f \leq F_{\lfloor \frac{k}{\tau} \rfloor} \leq \frac{F_0}{\frac{F_0 \lfloor \frac{k}{\tau} \rfloor}{C_\tau R^2} + 1} = \frac{F_0 C_\tau R^2}{F_0 \lfloor \frac{k}{\tau} \rfloor + C_\tau R^2},$$

which completes the proof. □

Remark 2 We consider a standard stepsize $\gamma = \frac{1}{L}$ and compare random BCD and MC-BCD. In [Theorem 1, [15]], it is shown that random BCD has the rate $\mathbb{E} f(x^k) - \min f = O(\frac{N \cdot R^2 \cdot L}{k})$. We stress that, with our notation, ∇f is $(N \cdot L)$-Lipschitz continuous in the worst case. When our Markov chain uses a complete graph, we can have a uniform stationary distribution and $\tau = 1$. In this case, MC-BCD reduces to random BCD, and our complexity of MC-BCD is also $\mathbb{E} f(x^k) - \min f = O(\frac{N \cdot R^2 \cdot L}{k})$. In this sense, we have generalized random BCD with a matching complexity. If the Markov chain promises a geometric mixing rate, i.e., $\tau = O(\ln N)$, then our convergence rate result becomes $\mathbb{E} f(x^k) - \min f = O(\frac{N \cdot \ln^2 N \cdot R^2 \cdot L}{k})$. While in

cyclic BCD, we have $f(x^k) - \min f = O(\frac{N^2 \cdot R^2 \cdot L}{k})$ from [2, Corollary 3.8].[2] That is, in terms of worst-case guarantee, MC-BCD performs slightly worse than i.i.d. random BCD but better than cyclic BCD. □

3.2.2 Linear convergence rate

To have linear convergence, we consider the restricted ν-strongly convex function:

$$f(x) - \min f \geq \nu \|x - \overline{x}\|^2, \quad \text{for all } x \in \mathbb{R}^N, \ \overline{x} = \text{Proj}_{\text{argmin } f}(x). \tag{40}$$

Theorem 3 *Under Assumptions 1 and 2, let $(x^k)_{k\geq 0}$ be generated by MC-BCD (8b) with $0 < \gamma < \frac{2}{L}$. If f satisfies condition (40), then*

$$\mathbb{E} f(x^k) - \min f \leq F_0 \left(1 - \frac{\nu}{C_\tau}\right)^{\lfloor \frac{k}{\tau} \rfloor}.$$

Proof Immediately from (40), we have the bound

$$\mathbb{E}\|x^{t\cdot\tau} - \overline{x^{t\cdot\tau}}\|^2 \leq \frac{\mathbb{E} f(x^{t\cdot\tau}) - \min f}{\nu} = \frac{F_t}{\nu}.$$

Substituting the above inequality into (31) yields $F_t^2 \leq \frac{C_\tau}{\nu} \cdot (F_t - F_{t+1}) \cdot F_t$, or equivalently $F_{t+1} \leq (1 - \frac{\nu}{C_\tau}) F_t$. Hence,

$$F_t \leq F_0 \left(1 - \frac{\nu}{C_\tau}\right)^t, \ \forall t \geq 0.$$

Again from monotonicity of $f(x^k)$ about k, it follows that

$$\mathbb{E} f(x^k) - \min f \leq F_{\lfloor \frac{k}{\tau} \rfloor} \leq F_0 \left(1 - \frac{\nu}{C_\tau}\right)^{\lfloor \frac{k}{\tau} \rfloor},$$

which completes the proof. □

Remark 3 If we consider the stepsize $\gamma = \frac{1}{L}$ and assume the Markov chain enjoys a uniform stationary distribution, then we get the rate $\mathbb{E}(f(x^k) - \min f) = O\left(\left(1 - \frac{\nu}{N \cdot \max\{8\kappa L \cdot (\tau - 1), 8L\}}\right)^{\lfloor \frac{k}{\tau} \rfloor}\right)$.

[2] The authors in [2, Corollary 3.8] present this results in the perspective of epochs, while here we present the rate in the perspective of iterations. Thus, their result is multiplied by N for comparison.

4 Extension to nonsmooth problems

All results established in previous sections assume the smoothness of the objective function. In this section, we add separable, possibly nonsmooth functions to the objective:

$$\text{minimize} \;\; F(x) \equiv f(x_1, x_2, \ldots, x_N) + \sum_{i=1}^{N} g_i(x_i). \tag{41}$$

Here, $f : \mathbb{R}^N \mapsto \mathbb{R}$ is a differentiable function, $\nabla_i f$ is Lipschitz continuous for each $i = 1, 2, \ldots, N$, and $g_i : \mathbb{R} \mapsto \mathbb{R}$ is a closed proper function. Note that we do not assume convexity on either f or g_i's. Toward finding a solution to (41), we propose the inexact Markov chain proximal block coordinate descent (iMC-PBCD).

Given a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$, the iMC-PBCD iteratively performs:

$$\begin{aligned} &\text{sample } i_k \in \{j : (i_{k-1}, j) \in \mathscr{E}\} \sim P_{i_{k-1},j}(k), \\ &\text{compute } x_j^{k+1} = \begin{cases} \mathbf{Prox}_{\gamma g_j}\left(x_j^k - \gamma\left(\nabla_j f(x^k) + \epsilon^k\right)\right), & \text{if } j = i_k, \\ x_j^k, & \text{if } j \neq i_k. \end{cases} \end{aligned} \tag{42}$$

In the above update, γ is a step size, ϵ^k denotes the error in evaluating the partial gradient, and $\mathbf{Prox}_{\psi}(y)$ is the proximal mapping of a closed function ψ at y, defined as

$$\mathbf{Prox}_{\psi}(y) \in \operatorname*{argmin}_x \left\{ \psi(x) + \frac{1}{2}\|x - y\|^2 \right\}.$$

To characterize the property of a solution, we employ the notion of subdifferential [23, Definition 8.3].

Definition 3 *(Subdifferential)* Let $J : \mathbb{R}^N \to (-\infty, +\infty]$ be a proper and lower semicontinuous function.

1. For any $x \in \mathrm{dom}(J)$, the Fréchet subdifferential of J at x, denoted as $\hat{\partial} J(x)$, is the set of all vectors $u \in \mathbb{R}^N$ that satisfies

$$\lim_{y \neq x} \inf_{y \to x} \frac{J(y) - J(x) - \langle u, y - x \rangle}{\|y - x\|} \geq 0.$$

   If $x \notin \mathrm{dom}(J)$, then $\hat{\partial} J(x) = \emptyset$.
2. The limiting subdifferential, or simply the subdifferential, of J at $x \in \mathrm{dom}(J)$, denoted as $\partial J(x)$, is defined as

$$\begin{aligned} \partial J(x) := \{u \in \mathbb{R}^N : \exists\, (x^k)_{k \geq 0} \text{ and } u^k \in \hat{\partial} J(x^k) \text{ such that } J(x^k) \\ \to J(x) \text{ and } u^k \to u \text{ as } k \to \infty\}. \end{aligned}$$

The first-order optimality condition for x to be a solution of (41) is

$$\mathbf{0} \in \partial F(x).$$

Any such point is called a critical point of F.

The proofs below are quite different from previous ones because we cannot bound the gradient with $\|\Delta^k\|$ any more, i.e., the core relation (20) fails to hold. Consequently, the convergence result in this section is new. Also, we cannot specify the convergence rates yet.

Lemma 5 *Under Assumption 2, let $(x^k)_{k\geq 0}$ be generated by iMC-PBCD (42) with $0 < \gamma < \frac{1}{L}$. If $\sum_{k=0}^{\infty} \|\epsilon^k\|^2 < \infty$, then*

$$\lim_{k\to\infty} \Delta^k = \boldsymbol{0}, \tag{43}$$

where Δ^k is defined in (9).

Proof By the definition of the proximal mapping, the update in (42) can be equivalently written as

$$x_{i_k}^{k+1} \in \operatorname*{argmin}_{x_{i_k}} \left\{ \left\langle x_{i_k} - x_{i_k}^k, \left(\nabla_{i_k} f(x^k) + \epsilon^k\right)\right\rangle + \frac{1}{2\gamma}\|x_{i_k} - x_{i_k}^k\|^2 + g_{i_k}(x_{i_k}) \right\}. \tag{44}$$

Therefore,

$$\left\langle x_{i_k}^{k+1} - x_{i_k}^k, \nabla_{i_k} f(x^k) + \epsilon^k \right\rangle + \frac{1}{2\gamma}\|x_{i_k}^{k+1} - x_{i_k}^k\|^2 + g_{i_k}(x_{i_k}^{k+1}) \leq g_{i_k}(x_{i_k}^k). \tag{45}$$

By the Young's inequality and the definition of Δ^k in (9), it holds that

$$\left\langle x_{i_k}^{k+1} - x_{i_k}^k, \epsilon^k \right\rangle \leq \frac{1}{4}\left(\frac{1}{\gamma} - L\right)\|\Delta^k\|^2 + \frac{\|\epsilon^k\|^2}{\frac{1}{\gamma} - L}.$$

In addition, it follows from (15) that

$$f(x^{k+1}) \leq f(x^k) + \left\langle \Delta^k, \nabla f(x^k)\right\rangle + \frac{L}{2}\|\Delta^k\|^2.$$

Adding the above two inequalities into (45) and recalling the definition of Δ^k in (9) give

$$f(x^{k+1}) + g_{i_k}(x_{i_k}^{k+1}) + \frac{1}{2\gamma}\|\Delta^k\|^2 \leq f(x^k) + g_{i_k}(x_{i_k}^k) + \frac{1}{4}\left(\frac{1}{\gamma} - L\right)\|\Delta^k\|^2 + \frac{\|\epsilon^k\|^2}{\frac{1}{\gamma} - L} + \frac{L}{2}\|\Delta^k\|^2.$$

Rearranging terms of the above inequality and noting $g_j(x_j^{k+1}) = g_j(x_j^k)$ for all $j \neq i_k$, we have

$$F(x^{k+1}) + \left(\frac{1}{4\gamma} - \frac{L}{4}\right)\|\Delta^k\|^2 \leq F(x^k) + \frac{\|\epsilon^k\|^2}{\frac{1}{\gamma} - L},$$

or equivalently

$$\frac{1}{4}\left(\frac{1}{\gamma} - L\right)\|\Delta^k\|^2 \leq F(x^k) - F(x^{k+1}) + \frac{\|\epsilon^k\|^2}{\frac{1}{\gamma} - L}.$$

Summing up the above inequality over k, using the conditions $0 < \gamma < \frac{1}{L}$ and $\sum_{k=0}^{\infty}\|\epsilon^k\|^2 < \infty$, and also noting F is lower bounded yield $\sum_{k=0}^{\infty}\|\Delta^k\|^2 < \infty$, which implies (43) and completes the proof. □

Theorem 4 *Under Assumptions 1 and 2, let $(x^k)_{k\geq 0}$ be generated by iMC-PBCD (42) with $0 < \gamma < \frac{1}{L}$. If $\sum_{k=0}^{\infty}\|\epsilon^k\|^2 < \infty$, then any cluster point of $(x^k)_{k\geq 0}$ is a critical point of F almost surely.*

Proof By the first optimality condition of (44), it holds

$$\frac{-\Delta_{i_k}^k}{\gamma} - \nabla_{i_k} f(x^k) - \epsilon^k \in \partial g_{i_k}(x_{i_k}^{k+1}),$$

or equivalently

$$-\frac{\Delta_{i_k}^k}{\gamma} + \nabla_{i_k} f(x^{k+1}) - \nabla_{i_k} f(x^k) - \epsilon^k \in \nabla_{i_k} f(x^{k+1}) + \partial g_{i_k}(x_{i_k}^{k+1}) = \partial_{i_k} F(x^{k+1}). \tag{46}$$

From (43) and also the Lipschitz continuity of $\nabla_i f$, we have from (46) that

$$\lim_{k\to\infty} \operatorname{dist}\left(0, \partial_{i_k} F(x^{k+1})\right) \leq \lim_{k\to\infty} \left\| -\frac{\Delta_{i_k}^k}{\gamma} + \nabla_{i_k} f(x^{k+1}) - \nabla_{i_k} f(x^k) - \epsilon^k \right\| = 0.$$

Let $\bar{x}$ be a cluster point of $(x^k)_{k\geq 0}$ and thus there is a subsequence $(x^k)_{k\in\mathscr{K}} \to \bar{x}$. If necessary, take a sub-subsequence out of $\mathscr{K}$, so without loss of generality, we can assume $|k_1 - k_2| \geq \tau$ for any $k_1, k_2 \in \mathscr{K}$. We go to prove the following claim:

$$\text{For any } j \in [N], \text{ there are infinite } k \in \mathscr{K} \text{ such that } i_k = j, \ \ a.s. \tag{47}$$

If the above claim is not true, then for some $j \in [N]$, with nontrivial probability, there are only finite $k \in \mathscr{K}$ such that $i_k = j$. Dropping these finitely many k's in $\mathscr{K}$,

we obtain a new subsequence $\hat{\mathscr{K}} = \{k_1, k_2, \ldots\}$ and $i_k \neq j$ for any $k \in \hat{\mathscr{K}}$. By the Markov property, it holds that for any $m \geq 1$,

$$\begin{aligned}&\mathbb{P}(i_{k_1} \neq j, i_{k_2} \neq j, i_{k_3} \neq j, \ldots, i_{k_m} \neq j)\\ &\quad = \mathbb{P}(i_{k_1} \neq j)\mathbb{P}(i_{k_2} \neq j \mid i_{k_1} \neq j)\mathbb{P}(i_{k_3} \neq j \mid i_{k_2} \neq j)\ldots\mathbb{P}(i_{k_m} \neq j \mid i_{k_{m-1}} \neq j).\end{aligned} \tag{48}$$

For any $k_{t-1}, k_t \in \hat{\mathscr{K}}$, since $k_t - k_{t-1} \geq \tau$, then we have from (11) that $\mathbb{P}(i_{k_t} = j \mid i_{k_{t-1}} \neq j) \geq \frac{\pi^*_{\min}}{2}$. Hence

$$\mathbb{P}(i_{k_t} \neq j \mid i_{k_{t-1}} \neq j) = 1 - \mathbb{P}(i_{k_t} = j \mid i_{k_{t-1}} \neq j) \leq 1 - \frac{\pi^*_{\min}}{2},$$

and thus it follows from (48) that

$$\mathbb{P}(i_{k_1} \neq j, i_{k_2} \neq j, i_{k_3} \neq j, \ldots, i_{k_m} \neq j) \leq \left(1 - \frac{\pi^*_{\min}}{2}\right)^{m-1}.$$

Letting $m \to \infty$, we conclude that

$$\mathbb{P}\big(\mathscr{K} \text{ only contains finitely many } k \text{ such that } i_k = j\big) = 0,$$

and thus the claim in (47) is true.

Now for any $j \in [N]$, taking $k \in \mathscr{K}$ such that $i_k = j$ and letting $k \to \infty$, we have from the fact $(x^{k+1})_{k\in\mathscr{K}} \to \bar{x}$ because of (43) and also the outer-continuity of subdifferential that

$$\operatorname{dist}(0, \partial_j F(\bar{x})) = \lim_{k\in\mathscr{K}, i_k = j} \operatorname{dist}(0, \partial_{i_k} F(x^{k+1})) = 0, \ a.s.$$

Therefore, we complete the proof. □

5 Empirical Markov chain dual coordinate ascent

In this section, we consider a special case of the risk minimization problem in form of (6). As we mentioned in section 1.1, if it is easy to get i.i.d. samples from the distribution Π of the sample space, then we can easily apply SDCA to (6). However, there are some cases where the distribution Π is not explicitly given and the samples are generated by a simulator, such as an MCMC sampler. Assume that the samples generated by the simulator form a Markov chain with stationary distribution Π. Generating i.i.d. samples may take very long time in this case, instead we want to make use of all the samples on a sample trajectory, which are not i.i.d. distributed.

Assume that the sample space Ξ is finite. Let $p_\xi \in (0, 1)$ denote the probability mass of $\xi \in \Xi$. Then, problem (6) can be presented as

$$\text{minimize}_{w\in\mathbb{R}^n} \sum_{\xi\in\Xi} p_\xi F(w^\top \xi) + \frac{\lambda}{2}\|w\|^2.$$

The objective function involves unknown parameters $(p_\xi)_{\xi\in\Xi}$. One way to solve this problem is to do the following two steps: first run the simulator for long enough time to get an estimation of $(p_\xi)_{\xi\in\Xi}$ (e.g. use frequency), denoted by $(\bar{p}_\xi)_{\xi\in\Xi}$; then minimize (6) with $(\bar{p}_\xi)_{\xi\in\Xi}$ by SDCA. The SDCA iteration in this case would be:

$$v^k = v^{k-1} + \frac{\alpha^k_{\xi^k}\xi^k}{\lambda} - \frac{\alpha^{k-1}_{\xi^k}\xi^k}{\lambda},$$
$$\alpha^{k+1}_{\xi^k} = \alpha^k_{\xi^k} - \gamma\Big((\xi^k)^\top v^k - \nabla F^*\big(\frac{-\alpha^k_{\xi^k}}{\bar{p}_{\xi^k}}\big)\Big),$$

where ξ^k is uniformly randomly chosen from Π, F^* is the conjugate function of F, $\alpha := (\alpha_\xi)_{\xi\in\Xi}$ are dual variables, and γ is the stepsize.

Compared with SDCA, the advantage of MC-DCA is to do sampling and minimization simultaneously. However, it still needs to estimate $(p_\xi)_{\xi\in\Xi}$. To address this issue, we introduce a practical way that approximates p_ξ by keeping a count $c_\xi(k)$, the times that sample ξ is chosen between iterations 1 and k. We estimate p_ξ by the sample frequency $c_\xi(k)/k$. We call it *empirical MC-DCA*. The empirical MC-DCA iteration is almost the same as SDCA iteration except that $(\xi^k)_{k\geq 0} \subseteq \Xi$ is a Markov chain and $\bar{p}_{\xi^k} = c_\xi(k)/k$.

We provide the theoretical performance of the empirical MC-DCA under a lower boundedness assumption on the frequency and geometric convergence of the Markov chain sampling.

Assumption 3 There exists a universal constant $\delta > 0$ such that for any $\xi \in \Xi$ and $k \in \mathbb{N}$, $c_\xi/k \geq \delta > 0$. In addition, there exists $0 < \lambda < 1$ such that, for any integer $l \in \mathbb{N}$, $|P(\xi^{k+l} = \xi|\xi^l = \xi) - p_\xi| = O(\lambda^k)$.

The time-homogeneous, irreducible, and aperiodic Markov Chain can satisfy Assumption 3.

Corollary 1 *Let $\alpha^k := (\alpha^k_\xi)_{\xi\in\Xi}$ be generated by the empirical MC-DCA and Assumption 3 hold. Then for the dual function given in (3), by denoting $A := \sup_{1\leq i\leq k,\xi\in\Xi}\{\|\alpha^k_\xi\|^2\}$, it holds that*

$$\mathbb{E}\big[\min_{1\leq i\leq k}\|\nabla D(\alpha^i)\|^2\big] = O\left(\frac{A\cdot\ln^2 k}{k}\right),$$

Proof Obviously, the empirical MC-DCA can be regarded as the inexact MC-BCD to minimize $D(\alpha)$ with the noise

$$e^k = \nabla F^*\Big(\frac{-\alpha_{\xi^k}^k}{p_\xi}\Big) - \nabla F^*\Big(\frac{-\alpha_{\xi^k}^k}{c_\xi(k)/k}\Big).$$

We have presented the convergence result of the inexact MC-BCD in Theorem 1. Thus, our work turns to bounding e^k. With Assumption 3,

$$\|e^k\|^2 = O\Big(A \cdot \frac{\|c_\xi(k) - k \cdot p_\xi\|^2}{k^2}\Big).$$

We now estimate the upper bound of $\mathbb{E}\|c_\xi(k) - k \cdot p_\xi\|^2$. Denote $\mathbf{1}_\xi(\cdot)$ as the variable valued as 1 when $\cdot = \xi$ and 0 when $\cdot$ being others. Then, $c_\xi(k)$ can be represented as

$$c_\xi(k) = \sum_{i=1}^{k} \mathbf{1}_\xi(\xi^i).$$

Direct calculation then gives

$$\begin{aligned}\mathbb{E}\|c_\xi(k) - k \cdot p_\xi\|^2 = \mathbb{E}\|\sum_{i=1}^{k} \mathbf{1}_\xi(\xi^i) - k \cdot p_\xi\|^2 = \underbrace{\sum_{i=1}^{k} \mathbb{E}\mathbf{1}_\xi^2(\xi^i)}_{a)} \underbrace{-2kp_\xi \sum_{i=1}^{k} \mathbb{E}\mathbf{1}_\xi(\xi^i)}_{b)} \\ \underbrace{+\,2\sum_{i<j} \mathbb{E}\big(\mathbf{1}_\xi(\xi^i)\mathbf{1}_\xi(\xi^j)\big)}_{c)} + k^2 p_\xi^2. \end{aligned} \tag{49}$$

With Assumption 3, we have

$$a) = kp_\xi + O\left(\sum_{i=1}^{k} \lambda^i\right) = kp_\xi + O\left(\frac{1}{1-\lambda}\right). \tag{50}$$

Similarly, we can derive

$$b) = -2k^2 p_\xi^2 + O\left(\frac{k}{1-\lambda}\right).$$

Now, we focus on bounding $c)$. The difficulty is the dependence of the variables. Denote the σ-algebra χ^k generated by $\xi^0, \xi^1, \ldots, \xi^k$, i.e., $\chi^k := \sigma(\xi^0, \xi^1, \ldots, \xi^k)$. Thus, we first derive the conditional expectation and then use the property $\mathbb{E}(\mathbb{E}(\cdot \mid \chi^i)) = \mathbb{E}(\cdot)$. Noting that $i < j$, we have

$$\mathbb{E}\big(\mathbf{1}_\xi(\xi^i)\mathbf{1}_\xi(\xi^j) \mid \chi^i\big) = \mathbb{P}(\xi^j = \xi \mid \xi^i = \xi) \cdot \mathbb{E}\big(\mathbf{1}_\xi(\xi^i) \mid \chi^i\big).$$

Taking expectations on both sides, we are then led to

$$\mathbb{E}\big(\mathbf{1}_\xi(\xi^i)\mathbf{1}_\xi(\xi^j)\big) = \mathbb{P}(\xi^j = \xi \mid \xi^i = \xi) \cdot \mathbb{E}\big(\mathbf{1}_\xi(\xi^i)\big) = \mathbb{P}(\xi^j = \xi \mid \xi^i = \xi) \cdot \mathbb{P}(\xi^i = \xi).$$

With the facts that $\mathbb{P}(\xi^j = \xi \mid \xi^i = \xi) = p_\xi + O(\lambda^{j-i})$ and $\mathbb{P}(\xi^i = \xi) = p_\xi + O(\lambda^i)$,

$$\mathbb{E}\big(\mathbf{1}_\xi(\xi^i)\mathbf{1}_\xi(\xi^j)\big) = p_\xi^2 + O(\max\{\lambda^i, \lambda^{j-i}\}).$$

Obviously, it holds

$$\sum_{i<j\le k} \max\{\lambda^i, \lambda^{j-i}\} = \sum_{t=1}^{\lceil \frac{k}{2} \rceil} c_t \lambda^t. \tag{51}$$

Now, we investigate what c_t exactly is. For any $1 \le t \le \lceil \frac{k}{2} \rceil$, λ^t only appears in the cases (I) $i = t$ and $j - i \ge t$ or (II) $j - i = t$ and $i \ge t$. Thus, we can get

$$c_t \le \sharp(\mathrm{I}) + \sharp(\mathrm{II}) = k - 2t + 1 + k - 2t + 1 = 2k - 4t + 2.$$

Thus, we derive

$$\sum_{i<j\le k} \max\{\lambda^i, \lambda^{j-i}\} \le \sum_{t=1}^{\lceil \frac{k}{2} \rceil} (2k - 4t)\lambda^t = O\left(\frac{k}{1-\lambda}\right). \tag{52}$$

With (51) and (52), we then get

$$c) = (k^2 - k)p_\xi + O\left(\frac{k}{1-\lambda}\right).$$

Substituting the bounds of (a), (b) and (c) to (49), we get

$$\mathbb{E}\|c_\xi(k) - k \cdot p_\xi\|^2 = O\left(\frac{k}{1-\lambda}\right).$$

Thus, the expectation of noise is bounded as

$$\mathbb{E}\|e^k\|^2 = O\left(\frac{A}{k(1-\lambda)}\right).$$

By a slight modification of the proof of inexact MC-BCD, we then prove the result. □

We also use a numerical experiment to verify the convergence of empirical MC-DCA and comparison with SDCA. We created a 40-state Markov chain with non-uniform stationary distribution. We randomly generated $x \in \mathbb{R}^{20}$, $\xi_i \in \mathbb{R}^{20}, i = 1, \ldots, 40$, and set $b_i = \xi_i^\top x$, where i is a state of the Markov chain. We also set

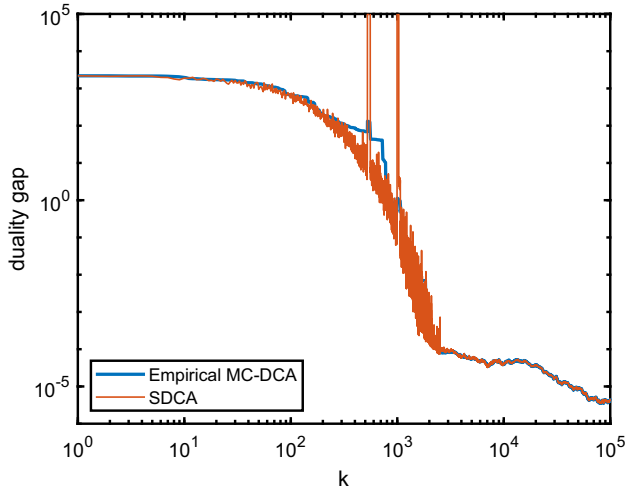

Fig. 1 Duality gap after k samples and k iterations of the algorithms. Empirical MC-DCA runs each iteration along with sampling. SDCA obtains all samples first and then runs k iterations with $(\bar{p}_\xi)_{\xi\in\Xi}$ estimated from the k samples

$F_i(x) = x - b_i$ and $\lambda = 0.1$. We compare duality gap of empirical MC-DCA and SDCA when doing the same number of samples and iterations. The MC-DCA runs each iteration along with sampling, while SDCA does sampling first and then does minimization with $(\bar{p}_\xi)_{\xi\in\Xi}$ estimated from the samples. Figure 1 shows that empirical MC-DCA can reach the same convergence rate as SDCA. However, empirical MC-DCA can minimize along sampling and does not need to store the sample space in memory. It can reach any accuracy as long as the sampling process continues. However, SDCA requires the knowledge of the sample space at each iteration. To improve the accuracy, it must resume the sampling process to re-estimate $(p_\xi)_{\xi\in\Xi}$.

6 Conclusion

In summary, we propose a new class of BCD method that can be implemented by visiting a random sequence of nodes in a network. As long as the network is connected, the method can run without the knowledge of its topology and other global parameters. Besides networks, our method can be also used for certain Markov decision processes. It can also run along with MCMC samples for empirical risk minimization when the underlying distribution cannot be sampled directly. The convergence of our method is proved for both convex and nonconvex objective functions with constant stepsize. Inexact subproblems are allowed. When the objective is convex and strongly convex, sublinear and linear convergence rates are proved, respectively.

References

1. Allen-Zhu, Z., Qu, Z., Richtárik, P., Yuan, Y.: Even faster accelerated coordinate descent using non-uniform sampling. In: *International Conference on Machine Learning (ICML)*, pp. 1110–1119 (2016)

2. Beck, A., Tetruashvili, L.: On the convergence of block coordinate descent type methods. SIAM J. Optim. **23**(4), 2037–2060 (2013)
3. Bradley, R.C., et al.: Basic properties of strong mixing conditions—a survey and some open questions. Probab Surv **2**, 107–144 (2005)
4. Brucker, P., Drexl, A., Möhring, R., Neumann, K., Pesch, E.: Resource-constrained project scheduling: notation, classification, models, and methods. Eur. J. Oper. Res. **112**(1), 3–41 (1999)
5. Chow, Y.T., Wu, T., Yin, W.: Cyclic coordinate-update algorithms for fixed-point problems: analysis and applications. SIAM J. Sci. Comput. **39**(4), A1280–A1300 (2017)
6. Dang, C., Lan, G.: Stochastic block mirror descent methods for nonsmooth and stochastic optimization. SIAM J. Optim. **25**(2), 856–881 (2015)
7. Fercoq, O., Richtárik, P.: Accelerated, parallel, and proximal coordinate descent. SIAM J. Optim. **25**(4), 1997–2023 (2015)
8. Hannah, R., Feng, F., Yin, W.: A2BCD: asynchronous acceleration with optimal complexity. In: *International Conference on Learning Representations (ICLR), New Orleans, LA* (2019)
9. Johansson, B., Rabi, M., Johansson, M.: A simple peer-to-peer algorithm for distributed optimization in sensor networks. In: *2007 46th IEEE Conference on Decision and Control*, pp. 4705–4710. IEEE (2007)
10. Lee, Y.T., Sidford, A.: Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems. In: *2013 IEEE 54th Annual Symposium on Foundations of Computer Science, Berkeley, CA, USA*, pp. 147–156. IEEE (2013)
11. Li, Y., Osher, S.: Coordinate descent optimization for ℓ^1 minimization with application to compressed sensing; a greedy algorithm. Inverse Probl. Imaging **3**(3), 487–503 (2009)
12. Li, Z., Uschmajew, A., Zhang, S.: On convergence of the maximum block improvement method. SIAM J. Optim. **25**(1), 210–233 (2015)
13. Liu, J., Wright, S.J.: Asynchronous stochastic coordinate descent: parallelism and convergence properties. SIAM J. Optim. **25**(1), 351–376 (2015)
14. Meyn, S.P., Tweedie, R.L.: Markov Chains and Stochastic Stability. Springer, Berlin (2012)
15. Nesterov, Y.: Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM J. Optim. **22**(2), 341–362 (2012)
16. Nutini, J., Schmidt, M., Laradji, I.H., Friedlander, M., Koepke, H.: Coordinate descent converges faster with the Gauss–Southwell rule than random selection. In: *International Conference on Machine Learning (ICML)*, pp. 1632–1641 (2015)
17. Peng, Z., Wu, T., Xu, Y., Yan, M., Yin, W.: Coordinate friendly structures, algorithms and applications. Ann. Math. Sci. Appl. **1**(1), 57–119 (2016)
18. Peng, Z., Xu, Y., Yan, M., Yin, W.: ARock: an algorithmic framework for asynchronous parallel coordinate updates. SIAM J. Sci. Comput. **38**(5), A2851–A2879 (2016)
19. Peng, Z., Xu, Y., Yan, M., Yin, W.: On the convergence of asynchronous parallel iteration with arbitrary delays. J. Oper. Res. Soc. China **1**(1), 5–42 (2019)
20. Peng, Z., Yan, M., Yin, W.: Parallel and distributed sparse optimization. In: *2013 Asilomar Conference On Signals, Systems and Computers*, pp. 659–646. IEEE (2013)
21. Ram, S.S., Nedić, A., Veeravalli, V.V.: Incremental stochastic subgradient algorithms for convex optimization. SIAM J. Optim. **20**(2), 691–717 (2009)
22. Richtárik, P., Takáč, M.: Parallel coordinate descent methods for big data optimization. Math. Program. **156**(1–2), 433–484 (2016)
23. Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis, vol. 317. Springer, Berlin (2009)
24. Shalev-Shwartz, S., Tewari, A.: Stochastic methods for l1-regularized loss minimization. J. Mach. Learn. Res. **12**(Jun), 1865–1892 (2011)
25. Shalev-Shwartz, S., Zhang, T.: Stochastic dual coordinate ascent methods for regularized loss minimization. J. Mach. Learn. Res. **14**(Feb), 567–599 (2013)
26. Sun, R., Hong, M.: Improved iteration complexity bounds of cyclic block coordinate descent for convex problems. In: *Advances in Neural Information Processing Systems*, pp. 1306–1314 (2015)
27. Sun, T., Hannah, R., Yin, W.: Asynchronous coordinate descent under more realistic assumptions. In: *Advances in Neural Information Processing Systems*, pp. 6183–6191 (2017)
28. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
29. Tseng, P., Yun, S.: A coordinate gradient descent method for nonsmooth separable minimization. Math. Program. **117**(1–2), 387–423 (2009)

30. Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J. Imaging Sci. **6**(3), 1758–1789 (2013)
31. Xu, Y., Yin, W.: Block stochastic gradient iteration for convex and nonconvex optimization. SIAM J. Optim. **25**(3), 1686–1716 (2015)
32. Yin, W., Mao, X., Yuan, K., Gu, Y., Sayed, A.H.: A communication-efficient random-walk algorithm for decentralized optimization. arXiv preprint arXiv:1804.06568 (2018)