Randomness and permutations in coordinate descent methods

Randomness and permutations in coordinate descent methods

Cite this article as: Mert Gürbüzbalaban, Asuman Ozdaglar, Nuri Denizcan Vanli and Stephen J. Wright, Randomness and permutations in coordinate descent methods, Mathematical Programming https://doi.org/10.1007/s10107-019-01438-4

Noname manuscript No.
(will be inserted by the editor)

Randomness and Permutations in Coordinate Descent Methods

**Mert Gürbüzbalaban, Asuman Ozdaglar,
Nuri Denizcan Vanli, Stephen J. Wright**

the date of receipt and acceptance should be inserted later

Abstract We consider coordinate descent (CD) methods with exact line search on convex quadratic problems. Our main focus is to study the performance of the CD method that use random permutations in each epoch and compare it to the performance of the CD methods that use deterministic orders and random sampling with replacement. We focus on a class of convex quadratic problems with a diagonally dominant Hessian matrix, for which we show that using random permutations instead of random with-replacement sampling improves the performance of the CD method in the worst-case. Furthermore, we prove that as the Hessian matrix becomes more diagonally dominant, the performance improvement attained by using random permutations increases. We also show that for this problem class, using any fixed deterministic order yields a superior performance than using random permutations. We present detailed theoretical analyses with respect to three different convergence criteria that are used in the literature and support our theoretical results with numerical experiments.

Keywords coordinate descent · random permutations · without-replacement sampling

M. Gürbüzbalaban
Rutgers University, Department of Management Science and Information Systems, 100 Rockafellar Road, Piscataway, NJ 08854, E-mail: mg1366@rutgers.edu

A. Ozdaglar
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, E-mail: asuman@mit.edu

N. D. Vanli
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, E-mail: denizcan@mit.edu

S. J. Wright
University of Wisconsin - Madison, Department of Computer Sciences and Wisconsin Institute for Discovery, 1210 West Dayton Street, Madison, WI 53706, E-mail: swright@cs.wisc.edu.

1 Introduction

We consider coordinate descent (CD) methods for solving unconstrained optimization problems of the form

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

where $f:\mathbb{R}^n\to\mathbb{R}$ is smooth and convex. CD methods have a long history in optimization [5,13,18] and have been used in many applications [10,16,20,22, 24]. They have seen a resurgence of recent interest because of their scalability and desirable empirical performance in machine learning and large-scale data analysis [3,26,31].

CD methods are iterative algorithms that perform (approximate) global minimizations with respect to a single coordinate (or several coordinates in the case of block CD) at each iteration. Specifically, at iteration k, an index $i_k \in \{1,2,\dots,n\}$ is chosen and the decision variable is updated to approximately minimize the objective function in the i_k-th coordinate direction (or at least to produce a significant decrease in the objective) [2,3]. The steps of this method are summarized in Algorithm 1, where $e_i = [0,\dots,0,1,0,\dots,0]^T$ is the i-th standard basis vector (with the i-th entry equal to one). At each iteration k, i_k-th coordinate of x is selected and a step is taken along the negative gradient direction in this coordinate. The counter $k=\ell n+j$ keeps track of the total number of iterations consisting of outer iterations indexed by ℓ and inner iterations indexed by the counter j. Each outer iteration is called a "cycle" or an "epoch" of the algorithm.

Algorithm 1 Coordinate Descent (CD)

Choose initial point $x^0\in\mathbb{R}^n$
for $\ell=0,1,2,\dots$ **do**
  for $j=0,1,2,\dots,n-1$ **do**
    Set $k=\ell n+j$
    Choose index $i_k=i(\ell,j)\in\{1,2,\dots,n\}$
    Choose stepsize $\alpha_k>0$
    $x^{k+1}\leftarrow x^k-\alpha_k[\nabla f(x^k)]_{i_k}e_{i_k}$, where $[\nabla f(x^k)]_{i_k}=e_{i_k}^T\nabla f(x^k)$
  end for
end for

CD methods use various schemes, both deterministic and stochastic, for choosing the coordinate i_k to be updated at iteration k. Prominent schemes include the following.

- Cyclic CD (CCD): The index $i(\ell,j)$ is chosen in a cyclic fashion over the elements in the set $\{1,2,\dots,n\}$ satisfying $i(\ell,j)=j+1$.
- Cyclic CD with a given order π (CCD-π): A permutation π of the set $\{1,2,\dots,n\}$ is selected. Then, the index $i(\ell,j)$ is chosen as the $(j+1)$-th element of π for every epoch ℓ. (CCD corresponds to the special case of $\pi=(1,2,\dots,n)$.)
- Randomized CD (RCD): The index $i(\ell,j)$ is chosen randomly with replacement from the set $\{1,2,\dots,n\}$ with uniform probabilities (each index has

the same probability of being chosen). This method is also known as the *stochastic CD* method.
- Random Permutations Cyclic CD (RPCD): At the beginning of each epoch ℓ, a permutation of $\{1, 2, \ldots, n\}$ is chosen, denoted by π_ℓ, uniformly at random over all permutations. Then, the index $i(\ell, j)$ is chosen as the $(j+1)$-th element of π_ℓ. Each permutation π_ℓ is independent of the permutations used at all previous and later epochs. This approach amounts to sampling indices from the set $\{1, 2, \ldots, n\}$ without replacement for each epoch.

While our focus in this paper will be on CD methods with the aforementioned selection rules, we note that several other variants of CD methods have been studied in the literature, including the Gauss-Southwell rule [17], in which i_k is selected in a greedy fashion to maximize $[\nabla f(x^k)]_i$, and versions of RCD [15], in which i_k is selected from a non-uniform distribution that may depend on the component-wise Lipschitz constants of f.

We are interested in the relative convergence behavior of these different variants of CD. While there have been some recent works that study and compare performances of CCD and RCD (for example, [1,9,15,23,27,28,30]); with the exception of a few recent papers (which focus on special quadratic problems, see [12,32]), there is limited understanding of the effects of random permutations in CD methods.

In this paper, we study convergence rate properties of RPCD for a special class of quadratic optimization problems with a diagonally dominant Hessian matrix, and compare its performance to that of RCD and CCD. Interest in RPCD is motivated by both empirical observations and practical implementation: In many machine learning applications, RPCD is observed numerically to outperform its with-replacement sampling counterpart RCD [14, 21]. Moreover, without-replacement sampling-based algorithms (such as RPCD and random reshuffling [4, 8]) are often easier to implement efficiently than their with-replacement counterparts (such as RCD and stochastic gradient descent) [12, 21] as it requires sequential data access, in contrast to the random data access required by with-replacement sampling (see e.g. [7, 25]).

We start by surveying briefly the existing results on the effects of random permutations for CD methods [12, 19, 28, 32]. Among these, Oswald and Zhou [19] studies the effects of random permutations on the convergence rate of the successive over-relaxation (SOR) method (that is used to solve linear systems) and presents a convergence rate on the expected function value of the iterates generated by the SOR method. The CD method, when applied to quadratic minimization problems, is equivalent to the SOR method (applied to the linear system that represents the first-order optimality condition of the quadratic problem) when the relaxation parameter is chosen as $\omega = 1$. Therefore, the convergence rate results in [19] readily extend for RPCD, when applied to quadratic problems. Sun and Ye [28] construct a quadratic problem, for which CCD requires $\mathcal{O}(n^2)$ times more iterations compared to RCD in order to achieve an ϵ-optimal solution (that is, a point x^k that satisfies $\mathbb{E}f(x^k) - f(x^*) \leq \epsilon$). For this problem, they also show that the distance of

the iterates (to the optimal solution) for CCD decays $\mathcal{O}(n^2)$ times slower than the distance of the expected iterates for RPCD and RCD. Lee and Wright [12] consider the same problem and present that the expected function values of RPCD and RCD decay with similar rates, while the asymptotic convergence rate of RPCD is shown to be slightly better than for RCD. In a following paper [32], the results in [12] are generalized to a larger class of quadratic problems through a more elaborate analysis.

Our main results provide convergence rate comparisons with respect to various criteria between RPCD, RCD, and CCD for a class of strongly convex quadratic optimization problems with a diagonally dominant Hessian matrix. In particular, we first provide an exact worst-case convergence rate comparison between RPCD, RCD, and CCD in terms of the distance of the expected iterates to the optimal solution, as a function of a parameter that represents the extent of diagonal dominance of the Hessian matrix. Our results show that, on this problem, CCD is always faster than RPCD, which in turn is always faster than RCD. Furthermore, we show that the relative convergence rate of RPCD to RCD goes to infinity as the Hessian matrix becomes more diagonally dominant. On the other extreme, as the Hessian matrix becomes less diagonally dominant, the ratio of convergence rates converges to a value in $[3/2, e-1)$, with the upper bound $e-1$ achieved in the limit as $n \to \infty$. Our second set of results compares the convergence rates of RPCD and RCD with respect to two other criteria that are widely used in the literature: the expected distance of the iterates to the solution and the expected function values of the iterates. For these criteria, we show that RPCD is faster than RCD in terms of the tightest upper bounds we obtain, and the amount of improvement increases as the matrices become more diagonally dominant.

The organization of the paper is as follows. In Section 2, we discuss the CCD, RCD, and RPCD algorithms in more detail and describe the three criteria that are used for analyzing convergence throughout the paper. In Section 3, we survey known results on the convergence rate of RPCD. We analyze the convergence rates of CCD, RCD, and RPCD with respect to the first convergence criterion in Section 4.1 and the behavior of RCD and RPCD with respect to the second and third convergence criteria in Section 4.2. We validate our theoretical results via numerical experiments in Section 5 and present conclusions in Section 6.

2 Preliminaries

To study performance of different CD methods, we focus on the special case of problem (1) when f is a strongly convex quadratic function:[1]

$$f(x) = \frac{1}{2} x^T A x, \tag{2}$$

[1] The results can be generalized for quadratic functions of the form $f(x) = \frac{1}{2}x^T Ax - b^T x$; however, for simplicity and compatibility with the earlier results in the literature, we consider the case $b = 0$.

where A is a positive definite matrix. We denote its extreme eigenvalues by

$$\mu := \lambda_{\min}(A) > 0, \quad L := \lambda_{\max}(A), \tag{3}$$

and note that μ is the modulus of convexity for f, while L is the Lipschitz constant for ∇f. The problem (1) has a unique solution $x^* = 0$ with optimal value $f(x^*) = 0$.

In the remainder of this section, we derive explicit formulas for the iterates of different variants of CD applied to (1) (in terms of matrix operators representing each epoch) and then introduce different convergence criteria for these variants. We show how asymptotic convergence rates can be characterized in terms of the spectral properties of A and the matrix operators for each epoch.

2.1 CD Methods

In this section, we describe the variants of the CD method (in particular, CCD, CCD-π, RCD, and RPCD) when applied to the quadratic problem in (2). The CD method (cf. Algorithm 1) with exact line search has the following update rule at each iteration

$$x^{k+1} = x^k - \frac{1}{A_{i_k i_k}}(Ax^k)_{i_k} e_{i_k}, \tag{4}$$

where the update coordinate i_k is determined according to one of the schemes mentioned above.

For the CCD algorithm, each coordinate is processed in a round-robin fashion using the standard cyclic order $(1, 2, \ldots, n)$. Denoting by D the diagonal part of A and by $-N$ the strictly lower triangular part of A, that is,

$$A = D - N - N^T,$$

the evolution of the iterates over an epoch (of n consecutive iterations) can be written as

$$x_{\text{CCD}}^{(\ell+1)n} = B_{\text{CCD}}\, x_{\text{CCD}}^{\ell n}, \quad \text{with} \quad B_{\text{CCD}} = (D - N)^{-1} N^T, \tag{5}$$

where ℓ denotes the epoch counter. Note that the update rule in (5) is equivalent to one iteration of the Gauss-Seidel method applied to the first-order optimality condition of (1), which is the linear system $Ax = 0$ (see Section 1.4 of [31] for details).

For the CCD-π algorithm, we let P_π denote the permutation matrix corresponding to order π and split the permuted Hessian matrix as follows:

$$A_\pi = P_\pi^T A P_\pi = D_\pi - N_\pi - N_\pi^T, \tag{6}$$

where $-N_\pi$ is a strictly lower triangular matrix and D_π is a diagonal matrix. Then, similar to (5), we have

$$x_{\text{CCD-}\pi}^{(\ell+1)n} = B_{\text{CCD-}\pi}\, x_{\text{CCD-}\pi}^{\ell n}, \quad \text{with} \quad B_{\text{CCD-}\pi} = (D_\pi - N_\pi)^{-1} N_\pi^T. \tag{7}$$

Note that B_{CCD} and $B_{\text{CCD-}\pi}$ are not symmetric matrices as the first column of both matrices are zero, whereas the first row contains nonzero entries.

For the RCD algorithm, the indices i_k are chosen independently at random at each iteration k. Denoting by x^k_{RCD} the k-th iterate generated by RCD, the update rule for RCD over a single iteration can be written as

$$x^{k+1}_{\text{RCD}} = B_{\text{RCD-}k}\, x^k_{\text{RCD}}, \quad \text{with} \quad B_{\text{RCD-}k} = I - \frac{1}{A_{i_k i_k}} e_{i_k} e_{i_k}^T A. \tag{8}$$

The expectation of $B_{\text{RCD-}k}$ with respect to the random variable i_k is denoted as follows:

$$B_{\text{RCD}} = \mathbb{E}_k B_{\text{RCD-}k}, \tag{9}$$

where we note that B_{RCD} is a symmetric matrix, by symmetry of A and uniform distribution of i_k.

For the RPCD algorithm, each coordinate is processed exactly once in each epoch according to a uniformly and independently chosen order. Recalling that π_ℓ denotes the permutation of coordinates used in epoch ℓ and using the iteration matrix corresponding to CCD-π_ℓ (see (7)), epoch ℓ of RPCD can be written as

$$x^{(\ell+1)n}_{\text{RPCD}} = B_{\text{RPCD-}\ell}\, x^{\ell n}_{\text{RPCD}}, \quad \text{with} \quad B_{\text{RPCD-}\ell} = P_{\pi_\ell} B_{\text{CCD-}\pi_\ell} P^T_{\pi_\ell}. \tag{10}$$

We introduce the following notation for the expected value of $B_{\text{RPCD-}\ell}$ with respect to permutation π_ℓ:

$$B_{\text{RPCD}} = \mathbb{E}_\ell B_{\text{RPCD-}\ell}, \tag{11}$$

where we note that B_{RPCD} is a symmetric matrix since π_ℓ is chosen uniformly at random over all permutations (see Lemma 1).

2.2 Convergence Rate Criteria

We next discuss how to measure and compare the convergence rates of different variants of CD. Three different improvement sequences have been used to measure the performance of CD methods in the literature:

$$\begin{aligned}
&(i) \quad \mathcal{I}_1(x^k_{\text{CD}}) = \left|\left|\mathbb{E}x^k_{\text{CD}} - x^*\right|\right|, && \text{(Distance of expected iterates)}\\
&(ii) \quad \mathcal{I}_2(x^k_{\text{CD}}) = \mathbb{E}\left|\left|x^k_{\text{CD}} - x^*\right|\right|^2, && \text{(Expected distance of iterates)}\\
&(iii) \quad \mathcal{I}_3(x^k_{\text{CD}}) = \mathbb{E}f(x^k_{\text{CD}}) - f(x^*). && \text{(Expected function value)}
\end{aligned}$$

(see e.g. [1, 9, 15, 22, 27, 28, 31]). While these three measures can be related to each other (Jensen's inequality yields $\mathcal{I}_1^2 \leq \mathcal{I}_2$ and strong convexity enables lower and upper bounding $\mathcal{I}_3$ between constant positive multiples of $\mathcal{I}_2$), we will provide different analyses for each of the measures to obtain the tightest estimates.

In the above definitions, expectations can be removed for deterministic algorithms such as CCD. By Jensen's inequality, we have that $\mathcal{I}_1^2(x^k_{\text{CD}}) \leq$

$\mathcal{I}_2(x^k_{\text{CD}})$ for all k. For a strongly convex function f, $\mathcal{I}_3$ can be lower and upper bounded between constant positive multiples of $\mathcal{I}_2$.

To study convergence rate of CCD, RCD, and RPCD with respect to improvement sequence $\mathcal{I}_1$, we use the operators derived in the previous section that represent one iterate or one epoch. The iteration matrices of CCD and RPCD are defined over an epoch (see (5) for CCD, (10) and (11) for RPCD). Therefore, using the generic subscript "CD" to represent the cases $B_{\text{CD}} = B_{\text{CCD}}$ for CCD and $B_{\text{CD}} = B_{\text{RPCD}}$ for RPCD, we have the following update rule

$$\mathbb{E}_\ell x^{(\ell+1)n}_{\text{CD}} = B_{\text{CD}}\, x^{\ell n}_{\text{CD}},$$

where $\mathbb{E}_\ell$ denotes the expectation with respect to the random variables in epoch ℓ given $x^{\ell n}_{\text{CD}}$. Note that the random variables in each epoch are independent and identically distributed across different epochs for RPCD (and RCD). Therefore, by using the law of iterated expectations, we obtain

$$\mathbb{E} x^{(\ell+1)n}_{\text{CD}} = B^\ell_{\text{CD}}\, x^0,$$

where $\mathbb{E}$ here denotes the expectation with respect to *all* random variables arising in the algorithm. Hence, the *worst-case convergence rate* with respect to $\mathcal{I}_1$ can be expressed as

$$\sup_{x^0 \in \mathbb{R}^n} \left(\frac{\left|\left| \mathbb{E} x^{\ell n}_{\text{CD}} \right|\right|}{||x^0||} \right)^{1/\ell} = \sup_{x^0 \in \mathbb{R}^n} \left(\frac{\left|\left| B^\ell_{\text{CD}}\, x^0 \right|\right|}{||x^0||} \right)^{1/\ell} = \left|\left| B^\ell_{\text{CD}} \right|\right|^{1/\ell}. \tag{12}$$

When B_{CD} is a symmetric matrix (as in RPCD), we have $\left|\left| B^\ell_{\text{CD}} \right|\right|^{1/\ell} = \rho(B_{\text{CD}})$. Hence, (12) yields a *per-epoch* worst-case convergence rate of $\rho(B_{\text{RPCD}})$ for RPCD. When B_{CD} is asymmetric (which is the case for CCD), we have by Gelfand's formula $\lim_{\ell \to \infty} \left|\left| B^\ell_{\text{CD}} \right|\right|^{1/\ell} = \rho(B_{\text{CD}})$. Thus, $\rho(B_{\text{CCD}})$ represents an *asymptotic* worst-case convergence rate measure for CCD.

For RCD, a similar derivation involving a single iteration (rather than one epoch) yields from (8) and (9) that

$$\mathbb{E}_k x^{k+1}_{\text{RCD}} = B_{\text{RCD}}\, x^k_{\text{CCD}}.$$

Similar reasoning to the above yields a *per-iteration* worst-case convergence rate of $\rho(B_{\text{RCD}})$, or equivalently a per-epoch rate of $\rho(B_{\text{RCD}})^n$, for RCD. (Note that, because B_{RCD} is symmetric, we have $\rho(B_{\text{RCD}}) = ||B_{\text{RCD}}||$.)

In our analysis of convergence rate of RCD with respect to improvement sequence $\mathcal{I}_2$, it follows from (8) that

$$\begin{aligned} \mathbb{E} \left|\left| x^{k+1}_{\text{RCD}} \right|\right|^2 &= (x^k_{\text{RCD}})^T \mathbb{E}\left[(B_{\text{RCD-}k})^T B_{\text{RCD-}k} \right] x^k_{\text{RCD}} \\ &\leq \left|\left| \mathbb{E}\left[(B_{\text{RCD-}k})^T B_{\text{RCD-}k} \right] \right|\right| \left|\left| x^k_{\text{RCD}} \right|\right|^2. \end{aligned}$$

For RPCD, we have similarly from (10) that

$$\begin{aligned}\mathbb{E}\left|\left|x_{\text{RPCD}}^{(\ell+1)n}\right|\right|^2 &= (x_{\text{RPCD}}^{\ell n})^T \mathbb{E}\left[(B_{\text{RPCD-}\ell})^T B_{\text{RPCD-}\ell}\right] x_{\text{RPCD}}^{\ell n} \\ &\leq \left|\left|\mathbb{E}\left[(B_{\text{RPCD-}\ell})^T B_{\text{RPCD-}\ell}\right]\right|\right| \left|\left|x_{\text{RPCD}}^{\ell n}\right|\right|^2.\end{aligned}$$

The matrices $\mathbb{E}\left[(B_{\text{RCD-}k})^T B_{\text{RCD-}k}\right]$ and $\mathbb{E}\left[(B_{\text{RPCD-}\ell})^T B_{\text{RPCD-}\ell}\right]$ are both symmetric. Convergence rates be obtained from $\rho\left(\mathbb{E}\left[(B_{\text{RCD-}k})^T B_{\text{RCD-}k}\right]\right)$ and $\rho\left(\mathbb{E}\left[(B_{\text{RPCD-}\ell})^T B_{\text{RPCD-}\ell}\right]\right)$ (or equivalently from the norms of these matrices), the first being a per-iteration convergence rate for RCD under criterion $\mathcal{I}_2$, and the second being a per-epoch rate for RPCD under the same criterion. Results along these lines appear in Section 4.2.

Finally, in our analysis of convergence rate of RCD with respect to $\mathcal{I}_3$, iteration (8) yields

$$\begin{aligned}\mathbb{E}f(x_{\text{RCD}}^{k+1}) &= (x_{\text{RCD}}^k)^T \mathbb{E}_k\left[(B_{\text{RCD-}k})^T A B_{\text{RCD-}k}\right] x_{\text{RCD}}^k \\ &= (A^{1/2} x_{\text{RCD}}^k)^T \mathbb{E}_k\left[A^{-1/2}(B_{\text{RCD-}k})^T A B_{\text{RCD-}k} A^{-1/2}\right] A^{1/2} x_{\text{RCD}}^k \\ &\leq \left|\left|\mathbb{E}_k\left[A^{-1/2}(B_{\text{RCD-}k})^T A B_{\text{RCD-}k} A^{-1/2}\right]\right|\right| \left|\left|A^{1/2} x_{\text{RCD}}^k\right|\right|^2.\end{aligned}$$

A similar analysis applied to the RPCD update formula (10) yields

$$\mathbb{E}f(x_{\text{RPCD}}^{(\ell+1)n}) \leq \left|\left|\mathbb{E}_\ell\left[A^{-1/2}(B_{\text{RPCD-}\ell})^T A B_{\text{RPCD-}\ell} A^{-1/2}\right]\right|\right| \left|\left|A^{1/2} x_{\text{RPCD}}^{\ell n}\right|\right|^2.$$

We will show that the matrices in these two bounds are symmetric. Thus, our convergence rate characterizations for RCD and RPCD with respect to $\mathcal{I}_3$ (see Section 4.2) will involve the norms (equivalently, the spectral radii) of these two matrices.

Remark 1 Note that for improvement sequence $\mathcal{I}_1$, the asymptotic worst-case convergence rate of the algorithm can be simply computed as the spectral radius of the expected iteration matrix. Furthermore, this bound is tight in the sense that there can be no smaller contraction rate c_1, for which an inequality of the type $\mathcal{I}_1(x_{\text{CD}}^{\ell n}) \leq c_1^\ell \mathcal{I}_1(x^0)$ asymptotically holds for all $x^0 \in \mathbb{R}^n$. Therefore, in Section 4.1, we compare the worst-case convergence rates of CCD, RCD and RPCD with respect to $\mathcal{I}_1$ through a tight analysis (in Proposition 2). We analyze the ratio of the convergence rates of RCD and RPCD in Proposition 1. On the other hand, for improvement sequences $\mathcal{I}_2$ and $\mathcal{I}_3$, we consider per-iteration and per-epoch upper bounds that are not necessarily asymptotically tight. Using a similar argument to (12), we can formulate the worst-case contraction factors for $\mathcal{I}_2$ and $\mathcal{I}_3$, but they would involve computation of powers of matrices (e.g., $\mathbb{E}\left[(B_{\text{CD-}k}^\ell)^T B_{\text{CD-}k}^\ell\right]$ and $\mathbb{E}\left[A^{-1/2}(B_{\text{CD-}k}^\ell)^T A B_{\text{CD-}k}^\ell A^{-1/2}\right]$), which does not admit a closed form characterization. Hence, in Section 4.2, we compare the convergence rates of RCD and RPCD based on per-iteration and per-epoch improvement rates, as has been done previously in the literature [12, 28, 32].

3 Prior work on CD methods with random permutations

In this section, we survey the known results on the performance of RPCD. There are several recent works that study the effects of random permutations in the convergence behavior of CD methods [12, 19, 28, 32]. To unify the randomization parameters (in RCD and RPCD) and the component-wise Lipschitz constants in different papers, we (without loss of generality) make the following assumption throughout the rest of the paper

$$A_{ii} = 1, \quad \text{for all} \quad i \in \{1, 2, \ldots, n\}. \tag{13}$$

This can always be satisfied by scaling the optimization variable, i.e., by setting $x = D^{-1/2}\tilde{x}$ in (2) and minimizing over $\tilde{x} \in \mathbb{R}^n$ (see e.g. [9, 32]).

Recently, Oswald and Zhou [19] analyzed the effects of random permutations for the successive over-relaxation (SOR) method, which is equivalent to the CD method with exact line search for a particular choice of algorithm parameter. They consider quadratic problems whose Hessian matrix is positive semidefinite and present convergence guarantees for SOR iterations with random permutations, which implies the following guarantee on the performance of RPCD.

Theorem 1 *[19, Theorem 4] Let f be a quadratic function of the form (2), where the Hessian matrix A has unit diagonals. Then, for any solution x^*, the RPCD algorithm enjoys the following guarantee*

$$\mathbb{E} f(x^{\ell n}_{RPCD}) - f(x^*) \leq \left(1 - \frac{\mu}{(1+L)^2}\right)^{\ell} \left(f(x^0) - f(x^*)\right). \tag{14}$$

Theorem 1 provides a convergence rate guarantee on the performance of RPCD for general quadratic functions. Under the same assumptions in Theorem 1, the best known upper bound on the performance of RCD is given by [15, Theorem 5]:

$$\mathbb{E}\left[\frac{1}{2}\left|\left|x^k_{\mathrm{RCD}} - x^*\right|\right|^2 + f(x^k_{\mathrm{RCD}}) - f(x^*)\right] \leq \left(1 - \frac{2\mu}{n(1+\mu)}\right)^k \left(\frac{1}{2}\left|\left|x^0 - x^*\right|\right|^2 + f(x^0) - f(x^*)\right). \tag{15}$$

This shows that the the upper bound on the performance of RCD per-epoch is approximately $\left(1 - \frac{2\mu}{n(1+\mu)}\right)^n \approx 1 - \frac{2\mu}{1+\mu}$, whereas it follows from (14) that the upper bound on the performance of RPCD can be as large as $1 - \frac{\mu}{(1+n)^2}$ since $L \leq \mathrm{tr}\,(A) = n$. These bounds suggest that RPCD may require $\mathcal{O}(n^2)$ times more iterations than RCD to guarantee an ϵ-optimal solution. However, empirical results show that RPCD often outperforms RCD in machine learning applications [6, 21]. Furthermore, it has been conjectured that the expected performance of RPCD should be no worse than the expected performance of RCD [21] (see also [11, 33] for related work on this conjecture). This motivates

to derive tight bounds for the convergence rate of RPCD and compare them with the known bounds on the convergence rate of RCD.

A similar phenomenon has been observed for CCD in comparison to RCD. In particular, the tightest known convergence rate results on the performance of CCD (see [1, 27, 28]) suggest that CCD may require $\widetilde{\mathcal{O}}(n^2)$ times more iterations than RCD to guarantee an ϵ-optimal solution. To understand this gap in the convergence rate bounds, Sun and Ye [28] focused on the quadratic problem in (2) with the following permutation invariant[2] Hessian matrix

$$A = \delta I + (1-\delta)\mathbf{1}\mathbf{1}^T, \quad \text{where} \quad \delta \in (0, n/(n-1)). \tag{16}$$

In particular, the authors considered a worst-case initialization and the case when δ is close to 0, for which $L = \mathcal{O}(n)$.[3] For this problem, they showed that CCD with the worst-case initialization indeed requires $\mathcal{O}(n^2)$ times more iterations than RCD to return an ϵ-optimal solution. They also provided rate comparisons between RPCD and CCD without providing a comparison between RPCD and RCD, which is presented in the following theorem.

Theorem 2 *[28, Proposition 3.4] Let $K_{\mathrm{CCD}}(\epsilon)$, $K_{\mathrm{RCD}}(\epsilon)$ and $K_{\mathrm{RPCD}}(\epsilon)$ be the minimum number of epochs for CCD, RCD and RPCD (respectively) to achieve (expected) relative error*

$$\frac{\|\mathbb{E}(x^k_{CD}) - x^*\|}{\|x^0 - x^*\|} \leq \epsilon,$$

for initial point $x^0 \in \mathbb{R}^n$ (for CCD, the expectation operator can be ignored). There exists a quadratic problem, whose Hessian matrix A satisfies (16) *for some δ around zero, such that*

$$\frac{K_{\mathrm{CCD}}(\epsilon)}{K_{\mathrm{RCD}}(\epsilon)} \geq \frac{n^2}{2\pi^2} \approx \frac{n^2}{20}, \tag{17a}$$

$$\frac{K_{\mathrm{CCD}}(\epsilon)}{K_{\mathrm{RPCD}}(\epsilon)} \geq \frac{n(n+1)}{2\pi^2} \approx \frac{n(n+2)}{20}. \tag{17b}$$

Theorem 2 shows that the worst-case performance (in improvement sequence $\mathcal{I}_1$) of RPCD and RCD is $\mathcal{O}(n^2)$ times faster than that of CCD. In a follow-up work, Lee and Wright [12] considered the same problem as [28] (see (16)) for the small δ case and presented asymptotic and non-asymptotic analyses of RPCD with respect to improvement sequence $\mathcal{I}_3$, presented in the following theorem.

Theorem 3 *[12, Theorem 3.3] Consider the quadratic problem* (2) *with the Hessian matrix A given by* (16)*, where $\delta \in (0, 0.4)$ and $n \geq 10$. For any $x^0 \in \mathbb{R}^n$, RPCD has the following non-asymptotic convergence guarantee*

$$\mathbb{E} f(x^{\ell n}_{RPCD}) - f(x^*) \leq (1 - 2\delta + 4\delta^2)^\ell R_0, \tag{18}$$

[2] A is a permutation invariant matrix if $PAP^T = A$, for any permutation matrix P.

[3] Since A has two eigenvalues: $\delta + n(1-\delta)$ with multiplicity 1 and δ with multiplicity $n-1$, the Lipschitz constant becomes $L = \delta + n(1-\delta)$, for $\delta \leq 1$; and as $\delta \to 0$, $L \to n$.

where R_0 is a constant depending on x_0 and δ. Furthermore, RPCD iterates enjoy an asymptotic convergence rate of

$$\lim_{\ell\to\infty}\left(\mathbb{E}f(x^{\ell n}_{RPCD})-f(x^*)\right)^{1/\ell}=1-2\delta-\frac{2\delta}{n}+2\delta^2+\mathcal{O}\left(\frac{\delta^2}{n}\right)+\mathcal{O}(\delta^3). \quad (19)$$

Theorem 3 shows that for the particular class of quadratic problems whose Hessian matrix satisfies (16), the convergence rate (in improvement sequence $\mathcal{I}_3$) of RPCD is faster than that of RCD in (15) in terms of the best known upper bounds (note that the convergence rate of RCD is approximately $1-2\delta/(1+\delta)$ for this case, see (15)). This is the first theoretical evidence that supports the empirical results showing RPCD often outperforms RCD [21]. In a follow-up work [32], Lee and Wright generalize the results of Theorem 3 to quadratic problems, whose Hessian matrix satisfies

$$A=\delta I+(1-\delta)uu^T, \quad \text{where} \quad \delta\in(0,n/(n-1)), \quad (20)$$

where $u\in\mathbb{R}^n$ is a vector with elements of size $\mathcal{O}(1)$ (this generalizes (16) that corresponds to $u=\mathbf{1}$). The conclusions are similar to [12], but the analysis is different because A is no longer a permutation-invariant matrix.

4 Performance of RPCD vs RCD on a class of diagonally dominant matrices

As described in the previous section, the existing works [12, 28] analyze the performance of RPCD for quadratic problems, whose Hessian satisfies (16) for small δ. Here, we consider the other extreme, i.e., the $\delta>1$ case, and provide tight convergence rate comparisons between RPCD, RCD and CCD with respect to all there improvement sequences defined in Section 2.2. In deriving convergence rate guarantees, we do not resort to the tools that are used in the earlier works on RPCD [12, 28, 32]. Instead, we present a novel analysis based on Perron-Frobenius theory that enables us to compute convergence rate bounds for all three criteria. For notational simplicity, we introduce the reformulation $\alpha=\delta-1$, which yields

$$A=(1+\alpha)I-\alpha\mathbf{1}\mathbf{1}^T, \quad \text{where} \quad \alpha\in(0,1/(n-1)). \quad (21)$$

It is simple to check that A has one eigenvalue at $1-(n-1)\alpha$ with the corresponding eigenvector $\mathbf{1}$ and other $n-1$ eigenvalues equal to $1+\alpha$. In particular, as α goes to zero, the condition number of A gets smaller and in the limit A is the identity matrix. On the other hand, as $\alpha\to\frac{1}{n-1}$, the matrix gets ill-conditioned. Therefore, the parameter

$$t:=\max_i\frac{\sum_{j\neq i}A_{ij}}{A_{ii}}=\alpha(n-1)\in(0,1) \quad (22)$$

is a measure of diagonal dominance. In the remainder of this section, we analyze the performance of RPCD, RCD and CCD in improvement sequence $\mathcal{I}_1$ and the performance of RPCD and RCD in improvement sequences $\mathcal{I}_2$ and $\mathcal{I}_3$ with respect to this diagonal dominance measure.

4.1 Convergence rates of RPCD, RCD and CCD in improvement sequence $\mathcal{I}_1$

In this section, we compare convergence rates of RPCD, RCD and CCD, where improvement sequence $\mathcal{I}_1(x^k) = \left|\left|\mathbb{E}x^k - x^*\right|\right|$ is chosen as the convergence criterion (as in Theorem 2). As we highlighted in Section 2.2, we first compute the expected iteration matrices of the RPCD and RCD algorithms, and show that they are symmetric. Then, we compute their spectral radii to conclude the per-epoch worst-case convergence rate of RPCD and RCD, and analyze their ratio in Proposition 1. We also show that the asymptotic worst-case convergence rate of CCD is faster than that of RPCD and RCD in Proposition 2.

We begin our discussion by writing the expected RPCD iterates (see (10) and (11)) as follows

$$\mathbb{E}_\ell x_{\text{RPCD}}^{(\ell+1)n} = B_{\text{RPCD}}\, x_{\text{RPCD}}^{\ell n}. \tag{23}$$

Note that since the Hessian matrix A is permutation invariant, the iteration matrix of the CCD-π algorithm for any cyclic order π is equal to the iteration matrix of the standart CCD algorithm, i.e., $B_{\text{CCD}} = B_{\text{CCD-}\pi}$ for all orders π. Therefore, we have $B_{\text{RPCD}} = \mathbb{E}_\pi[P_\pi B_{\text{CCD}} P_\pi^T] = \mathbb{E}_P[P B_{\text{CCD}} P^T]$, where we drop the subscript π from the matrices for notational simplicity. In order to obtain a formula for B_{RPCD}, we first reformulate the CCD iteration matrix in (5) as follows

$$B_{\text{CCD}} = (I-N)^{-1}N^T = I - (I-N)^{-1}(I-N-N^T) = I - \Gamma^{-1}A,$$

where $\Gamma = I - N$. Using this reformulation, the expected iteration matrix of RPCD can computed as follows

$$B_{\text{RPCD}} = \mathbb{E}_P\left[P B_{\text{CCD}} P^T\right] = \mathbb{E}_P\left[P(I-\Gamma^{-1}A)P^T\right] = I - \mathbb{E}_P\left[P\Gamma^{-1}P^T\right]A,$$

where we used the fact that $PP^T = I$ and $AP^T = P^T A$. For the case the Hessian matrix A satisfies (21), Γ^{-1} can be explicitly computed as

$$\Gamma^{-1} = \text{toeplitz}(c, r), \tag{24}$$

where $\text{toeplitz}(c,r)$ denotes the Toeplitz matrix with the first column c and the first row r, which are given by

$$c = \left[1,\, \alpha,\, \alpha(1+\alpha),\, \alpha(1+\alpha)^2,\, \ldots,\, \alpha(1+\alpha)^{n-2}\right]^T, \quad r = [1, 0, 0, \ldots, 0].$$

In order to compute $\mathbb{E}_P\left[P\Gamma^{-1}P^T\right]$, we use the following lemma, which states that expectation over all permutations separately averages the diagonal and off-diagonal entries of the permuted matrix.

Lemma 1 *[12, Lemma 3.1] Given any matrix $Q \in \mathbb{R}^{n\times n}$ and permutation matrix P selected uniformly at random from the set of all permutations, we have*

$$\mathbb{E}_P[PQP^T] = \tau_1 I + \tau_2 \mathbf{1}\mathbf{1}^T,$$

where

$$\tau_2 = \frac{\mathbf{1}^T Q \mathbf{1} - \text{trace}(Q)}{n(n-1)} \quad \textit{and} \quad \tau_1 = \frac{\text{trace}(Q)}{n} - \tau_2. \tag{25}$$

Letting $Q = \Gamma^{-1}$ in Lemma 1, we observe that the matrix $\mathbb{E}_P[P\Gamma^{-1}P^T]$ has diagonals equal to one and all the off-diagonal entries equal to each other:

$$\mathbb{E}_P[P\Gamma^{-1}P^T] = (1-\gamma)I + \gamma \mathbf{1}\mathbf{1}^T, \tag{26}$$

where γ can be found as the average of the off-diagonal entries of Γ^{-1}. The following lemma (whose proof is given in Appendix A) provides an explicit expression for γ.

Lemma 2 *For any $\alpha \in (0, 1/(n-1))$, we have*

$$\gamma = \frac{(1+\alpha)^n - \alpha n - 1}{\alpha n(n-1)},$$

where γ denotes the off-diagonal entries of $\mathbb{E}_P[P\Gamma^{-1}P^T]$ in (26).

Using Lemma 2, it follows from the definition of A in (21) and equation (26) that

$$B_{\mathrm{RPCD}} = I - \mathbb{E}_P[P\Gamma^{-1}P^T]A = ((n-1)\gamma - \beta)I + \beta \mathbf{1}\mathbf{1}^T,$$

where

$$\beta = \alpha - \gamma + \alpha\gamma(n-2).$$

Since B_{RPCD} is a symmetric matrix, then by (12), it suffices to compute the spectral radius of B_{RPCD} to obtain the worst-case performance of RPCD with respect to improvement sequence $\mathcal{I}_1$. To this end, we note that for any $\alpha \in (0, 1/(n-1))$, $B_{\mathrm{RPCD}} > 0$ since $B_{\mathrm{RPCD}} = \mathbb{E}_P[PB_{\mathrm{CCD}}P^T]$ and $B_{\mathrm{CCD}} \geq 0$ with at least one strictly positive entry in both the diagonal and off-diagonal parts (see also (47) for an explicit formula of B_{CCD}). Then, by the Perron-Frobenius Theorem [29, Lemma 2.8], we have

$$\begin{aligned}
\rho(B_{\mathrm{RPCD}}) &= \sum_{j=1}^{n} [B_{\mathrm{RPCD}}]_{ij}, \quad \text{for all } i \in [n] \\
&= (n-1)(\gamma\alpha + \beta) \\
&= (n-1)(\alpha - \gamma + \alpha\gamma(n-1)) \\
&= 1 - \left[(1-\alpha(n-1))\,(1+\gamma(n-1))\right].
\end{aligned}$$

Substituting the formula for γ from Lemma 2 above, we obtain the spectral radius of the RPCD iteration matrix as follows

$$\rho(B_{\mathrm{RPCD}}) = 1 - (1-\alpha(n-1))\,\frac{(1+\alpha)^n - 1}{\alpha n} = 1 - \frac{1-t}{n}\left(\frac{\left(1+\frac{t}{n-1}\right)^n - 1}{\frac{t}{n-1}}\right), \tag{27}$$

where $t = \alpha(n-1)$ denotes the diagonal dominance factor (as defined in (22)).

For the RCD algorithm, on the other hand, we have (by (8) and (9)) the following expected iterates

$$\mathbb{E}_k x^{k+1}_{\mathrm{RCD}} = B_{\mathrm{RCD}}\, x^k_{\mathrm{RCD}}, \quad \text{where} \quad B_{\mathrm{RCD}} = I - \frac{1}{n}A.$$

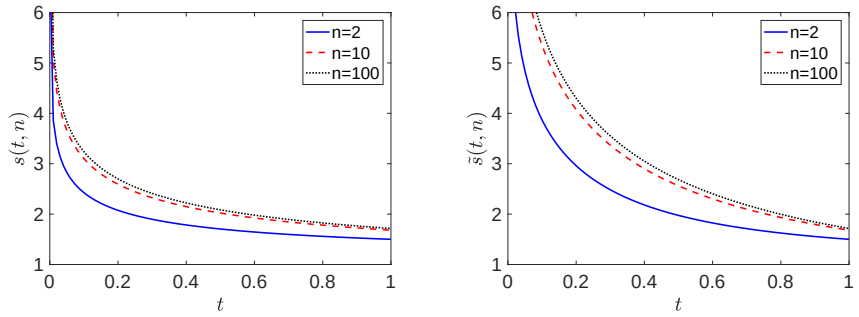

Fig. 1: Plot of $s(t,n)$ and $\tilde{s}(t,n)$ versus $t \in (0,1)$ for different values of n.

Since A is a symmetric matrix, then by (12), the per-epoch worst-case asymptotic rate of RCD with respect to improvement sequence $\mathcal{I}_1$ can be found as

$$\rho(B_{\mathrm{RCD}})^n = \left(1 - \frac{1}{n}\lambda_{\min}(A)\right)^n = \left(1 - \frac{1-t}{n}\right)^n.$$

In Proposition 1, we compare the performance of RPCD and RCD with respect to improvement sequence $\mathcal{I}_1$. To this end, we define

$$s(t,n) = \frac{-\log \rho(B_{\mathrm{RPCD}})}{-\log \rho(B_{\mathrm{RCD}})^n}, \tag{28}$$

(where log denotes the natural logarithm), which is equal to the ratio between the number of epochs required to guarantee $\left|\left|\mathbb{E}x^{\ell n} - x^*\right|\right| \leq \epsilon$ for RCD and RPCD algorithms. In particular $s(t,n) > 1$ implies RPCD has a faster worst-case convergence rate than RCD. In the following theorem, we show that RPCD is faster than RCD for any $t \in (0,1)$ and $n \geq 2$, and quantify the rate of improvement.

Proposition 1 *The following statements are true:*

(*i*) *The function $s(t,n)$ is strictly decreasing in t over $(0,1)$.*
(*ii*) $\lim_{t\to 0} s(t,n) = \infty$.
(*iii*) *Let $g(n) := \lim_{t\to 1} s(t,n)$. We have $g(n) \in [3/2, e-1)$, for any $n \geq 2$. Furthermore, $g(n)$ is strictly increasing in $n \geq 2$ satisfying*

$$g(2) = 3/2 \quad \textit{and} \quad \lim_{n\to\infty} g(n) = e - 1.$$

A consequence of Proposition 1 is that RPCD is faster than RCD in the worst-case, for every $t \in (0,1)$ by a factor $s(t,n) > 1$. Furthermore, the amount of acceleration $s(t,n)$ goes to infinity as $\alpha \to 0$ for any n fixed. This shows that as the matrix A becomes more and more well-conditioned (as $\alpha \to 0$), the amount of speed-up $s(t,n)$ we obtain with RPCD with respect to RCD goes to infinity. This is consistent with the observation that cyclic orders work well for diagonal-like matrices that are well-conditioned (see e.g. [29]). Proposition

1 is illustrated in Figure 1 (left panel), where we plot the parameter $s(t,n)$ as a function of t for different values of n.

We next compare the convergence rate of CCD with respect to RPCD and RCD. To this end, as we discuss in Section 2.2 (cf. (12)), we use $\rho(B_{\mathrm{CCD}})$ as the asymptotic per epoch worst-case convergence rate of CCD, whereas for comparison to RCD, we use a per-epoch rate of $\rho(B_{\mathrm{RCD}})^n$. Note that as discussed in (23), $B_{\mathrm{CCD}} = B_{\text{CCD-}\pi}$ for all π, and hence $\rho(B_{\mathrm{CCD}}) = \rho(B_{\text{CCD-}\pi})$ for all π. Although, explicit calculation of $\rho(B_{\mathrm{CCD}})$ appears to be challenging, we prove that the known upper bounds [9, Theorem 4.12] on $\rho(B_{\mathrm{CCD}})$ is tighter than $\rho(B_{\mathrm{RPCD}})$, which together with Proposition 1 imply the following result.

Proposition 2 *Let f be a quadratic function of the form* (2)*, whose Hessian matrix given by* (21)*. Then, the expected iteration matrices of CCD, RPCD and RCD satisfy*

$$\rho(B_{CCD}) < \rho(B_{RPCD}) < \rho(B_{RCD})^n, \tag{29}$$

for any $\alpha \in (0, 1/(n-1))$ and $n \geq 2$.

4.2 Convergence rates of RPCD and RCD in improvement sequences $\mathcal{I}_2$ & $\mathcal{I}_3$

In this section, we compare the rate of RPCD and RCD with respect to improvement sequences $\mathcal{I}_2$ and $\mathcal{I}_3$. When the Hessian matrix A satisfies (21), the smallest eigenvalue of A can be found as follows

$$\mu = 1 - t = 1 - \alpha(n-1). \tag{30}$$

Plugging this value in the convergence guarantee of RCD in (15), we can obtain a convergence guarantee on both improvement sequences $\mathcal{I}_2$ and $\mathcal{I}_3$ as the left hand-side of (15) upper bounds both $2\mathcal{I}_2$ and $\mathcal{I}_3$. However, for the particular problem class we consider in this paper, we derive a tighter convergence rate guarantee for RCD in the next proposition, whose proof is deferred to Appendix D.

Proposition 3 *Let f be a quadratic function of the form* (2)*, whose Hessian matrix given by* (21)*. Then, RCD iterations satisfy*

$$\mathbb{E}\|x^k_{RCD} - x^*\|^2 \leq \left(1 - \frac{2\mu}{n} + \frac{\mu^2}{n}\right)^k \|x^0 - x^*\|^2, \tag{31}$$

and

$$\mathbb{E}\left(f(x^k_{RCD}) - f(x^*)\right) \leq \left(1 - \frac{\mu}{n}\right)^k \left(f(x^0) - f(x^*)\right). \tag{32}$$

Remark 2 We observe that the upper bound in (31) is smaller (tighter) than the upper bound in (15) for any $\alpha \in (0, 1/(n-1))$ because

$$1 - \frac{2\mu}{n} + \frac{\mu^2}{n} < 1 - \frac{2\mu}{n} + \frac{2\mu^2}{n} = 1 - \frac{2\mu(1-\mu)}{n} = 1 - \frac{2\mu(1-\mu^2)}{n(1+\mu)} < 1 - \frac{2\mu}{n(1+\mu)},$$

where the inequalities are due to the fact that $\mu = 1 - \alpha(n-1) \in (0,1)$.

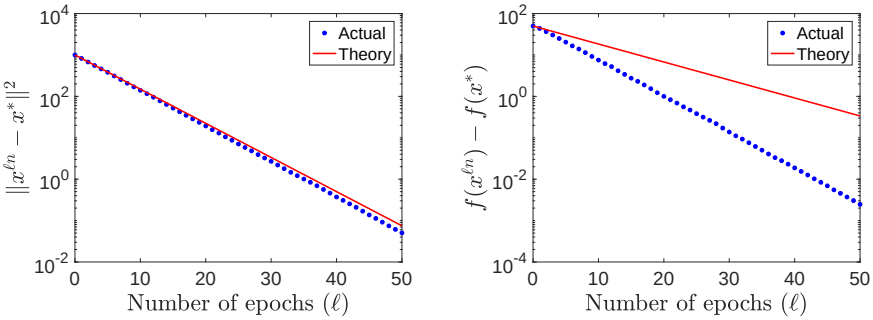

Fig. 2: Tightness of the bounds in Proposition 3 when $n = 1000$ and $\alpha = \frac{0.9}{n-1}$: Left figure for (31) and right figure for (32).

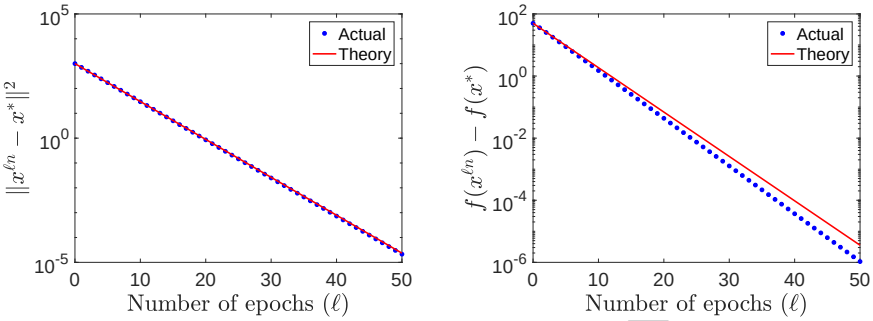

Fig. 3: Tightness of the bounds in Proposition 4 when $n = 1000$ and $\alpha = \frac{0.9}{n-1}$: Left figure for (33) and right figure for (34).

We next analyze the performance of RPCD in the following proposition and show that the convergence rate guarantee of RPCD is tighter than the convergence rate guarantee of RCD in Proposition 3. The proof of Proposition 4 is given in Appendix E.

Proposition 4 *Let f be a quadratic function of the form* (2)*, whose Hessian matrix given by* (21)*. Then, RPCD iterations satisfy*

$$\mathbb{E}\|x^{\ell n}_{RPCD} - x^*\|^2 \leq \left(1 - \frac{2\mu}{n}\left(\frac{(1+\alpha)^n - 1}{\alpha}\right) + \frac{\mu^2}{n}\left(\frac{(1+\alpha)^{2n} - 1}{\alpha(\alpha+2)}\right)\right)^{\ell} \|x^0 - x^*\|^2, \tag{33}$$

and

$$\mathbb{E} f(x^{\ell n}_{RPCD}) - f(x^*) \leq \left(1 - \frac{\mu}{n}\left(\frac{(1+\alpha)^{2n} - 1}{\alpha(\alpha+2)}\right)\right)^{\ell} \left(f(x^0) - f(x^*)\right). \tag{34}$$

We next compare the convergence rates we derive for the RCD and RPCD algorithms. In particular, we consider the convergence rate of both algorithms

in improvement sequence $\mathcal{I}_2$ since we obtain tighter upper bounds for it. Comparing the convergence rate bounds for RCD and RPCD in (31) and (33), respectively, we can observe that RPCD is faster (in terms of the best known rate guarantees) than RCD by a factor of

$$\tilde{s}(t,n) := \frac{-\log\left(1-\frac{2\mu}{n}\left(\frac{(1+\alpha)^n-1}{\alpha}\right)+\frac{\mu^2}{n}\left(\frac{(1+\alpha)^{2n}-1}{\alpha(\alpha+2)}\right)\right)}{-n\log\left(1-\frac{2\mu}{n}+\frac{\mu^2}{n}\right)},$$

which is plotted in Figure 1 (right panel) in the interval $t \in (0,1)$ for different values of n. We observe from this figure that the convergence rate bound for RPCD is better than than the one for RCD for all $t \in (0,1)$ and $n \geq 2$. Furthermore, the difference in convergence rate bounds increases as t gets smaller, i.e., as the Hessian matrix becomes more diagonally dominant. We can also show that $\tilde{s}(t,n)$ behaves similar to $s(t,n)$ as $t \to 1$, where the limiting values can be found in Proposition 1.

5 Numerical Experiments

Here we compare the performance of CCD, RPCD, and RCD for the quadratic problem (2) with Hessian matrix (21). In Figure 4, we use a worst-case initialization $x^0 = \mathbf{1}$, for $n \in \{1000, 10000\}$ and $\alpha \in \left\{\frac{0.01}{n-1}, \frac{0.50}{n-1}, \frac{0.99}{n-1}\right\}$. We observe that CCD is the faster than RPCD, which is faster than RCD. This behavior is in accordance with the theoretical results in Propositions 2-4. Furthermore, as α decreases, we can see that the ratio between the convergence rates of RPCD and RCD increases, consistent with Proposition 1 (see also Figure 1). We can also observe from the right column in Figure 4 that when α is close to $1/(n-1)$, the ratio between the convergence rates of RPCD and RCD is close to the theoretical limits obtained in Proposition 1 (see part *(iii)*, which shows that the ratio is in the interval $[3/2, e-1)$). Figure 5 plots similar results to Figure 4, but for a random initialization rather than worst-case initialization. Convergence rates depicted in Figure 5 are similar to those of Figure 4, due to the fact that $x^{\ell n}$ becomes colinear with the vector of ones as ℓ increases (as $\mathbf{1}$ is the leading eigenvector of the expected iteration matrix), so that the worst-case convergence rate dictates the performance of the algorithms.

6 Conclusion

In this paper, we surveyed the known results on the performance of RPCD for special cases of strongly convex quadratic objectives and add to these results by presenting a class of convex quadratic problems with diagonally dominant Hessians. Using the distance of the expected iterates to the optimal solution as the convergence criterion, we compared the ratio between the performances of RPCD and RCD with respect to a parameter that represents the extent of diagonal dominance. We illustrated that as the Hessian matrix becomes more diagonally dominant, this ratio goes to infinity, whereas as it gets smaller it

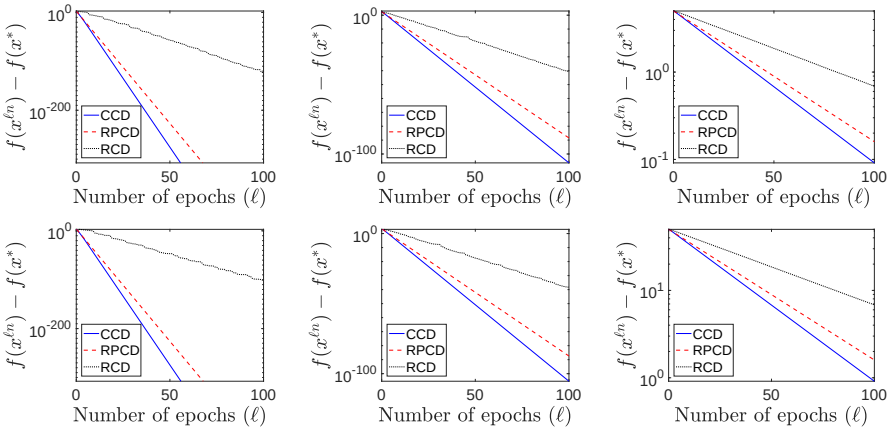

Fig. 4: CCD vs RPCD vs RCD with worst-case initialization for $n = 1000$ (top row) and $n = 10000$ (bottom row): $\alpha = \frac{0.01}{n-1}$ in the left column, $\alpha = \frac{0.50}{n-1}$ in the middle column, and $\alpha = \frac{0.99}{n-1}$ in the right column.

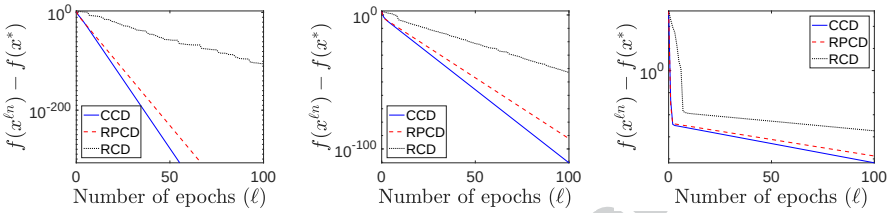

Fig. 5: CCD vs RPCD vs RCD with random initialization for $n = 1000$: $\alpha = \frac{0.01}{n-1}$ (left figure), $\alpha = \frac{0.50}{n-1}$ (middle figure), and $\alpha = \frac{0.99}{n-1}$ (right figure).

goes to a constant in the interval $[3/2, e-1)$. We also showed that CCD outperforms both RPCD and RCD for this class of problems. When expected distance of the iterates or expected function value of the iterates is used as the convergence criterion, we presented that the worst-case convergence rate bounds derived for RPCD are tighter compared to the ones for RCD. This is in accordance with our first set of results, i.e., when distance of the expected iterates is used as the convergence criterion. Computational experiments validate our theoretical results, which fill a gap between the theoretical guarantees for RPCD and its empirical performance.

7 Acknowledgements

Mert Gürbüzbalaban's research is supported in part by the grants NSF DMS-1723085 and NSF CCF-1814888.

References

1. A. Beck and L. Tetruashvili. On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization*, 23(4):2037–2060, 2013.
2. D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, 1999.
3. D. P. Bertsekas. *Convex Optimization Algorithms*. Athena Scientific, 2015.
4. D. P. Bertsekas. Incremental aggregated proximal and augmented lagrangian algorithms. *CoRR*, abs/1509.09257, 2015.
5. D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, Inc., 1989.
6. L. Bottou. Curiously fast convergence of some stochastic gradient descent algorithms. In *Proceedings of the symposium on learning and data science, Paris*, 2009.
7. L. Bottou. Stochastic gradient descent on toy problems, September 2012. http://leon.bottou.org/projects/sgd.
8. M. Gürbüzbalaban, A. Ozdaglar, and P. A. Parrilo. Why random reshuffling beats stochastic gradient descent. *arXiv:1510.08560*, 2015.
9. M. Gürbüzbalaban, A. Ozdaglar, P. A. Parrilo, and N. D. Vanli. When cyclic coordinate descent outperforms randomized coordinate descent. In *Advances in Neural Information Processing Systems*, pages 7002–7010, 2017.
10. C.-J. Hsieh, H.-F. Yu, and I. S. Dhillon. Passcode: Parallel asynchronous stochastic dual co-ordinate descent. In *ICML*, volume 37, pages 2370–2379, 2015.
11. A. Israel, F. Krahmer, and R. Ward. An arithmetic-geometric mean inequality for products of three matrices. *Linear Algebra and its Applications*, 488:1 – 12, 2016.
12. C.-P. Lee and S. J. Wright. Random Permutations Fix a Worst Case for Cyclic Coordinate Descent. *ArXiv e-prints*, July 2016.
13. Z.-Q. Luo and P. Tseng. On the convergence of the coordinate descent method for convex differentiable minimization. *Journal of Optimization Theory and Applications*, 72(1):7–35, 1992.
14. D. Needell and J. A. Tropp. Paved with good intentions: Analysis of a randomized block Kaczmarz method. *Linear Algebra and its Applications*, 441:199 – 221, 2014.
15. Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.
16. Y. Nesterov and S.U. Stich. Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM Journal on Optimization*, 27(1):110–123, 2017.
17. J. Nutini, M. Schmidt, I. Laradji, M. Friedlander, and H. Koepke. Coordinate descent converges faster with the gauss-southwell rule than random selection. In *International Conference on Machine Learning*, pages 1632–1641, 2015.
18. J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.
19. P. Oswald and W. Zhou. Random reordering in sor-type methods. *Numerische Mathematik*, 135(4):1207–1220, 2017.
20. Z. Qin, K. Scheinberg, and D. Goldfarb. Efficient block-coordinate descent algorithms for the group lasso. *Mathematical Programming Computation*, 5(2):143–169, 2013.
21. B. Recht and C. Ré. Toward a noncommutative arithmetic-geometric mean inequality: Conjectures, case-studies, and consequences. *JMLR Workshop and Conference Proceedings*, 23:11.1–11.24, 2012.
22. P. Richtárik and M. Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016.
23. A. Saha and A. Tewari. On the nonasymptotic convergence of cyclic coordinate descent methods. *SIAM Journal on Optimization*, 23(1):576–601, 2013.
24. G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J. S. Pang. Decomposition by partial linearization: Parallel optimization of multi-agent systems. *IEEE Transactions on Signal Processing*, 62(3):641–656, Feb 2014.
25. Ohad Shamir. Without-replacement sampling for stochastic gradient methods. In *Advances in Neural Information Processing Systems*, pages 46–54, 2016.
26. H.-J. M. Shi, S. Tu, Y. Xu, and W. Yin. A Primer on Coordinate Descent Algorithms. *ArXiv:1610.00040*, 2016.

27. R. Sun and M. Hong. Improved iteration complexity bounds of cyclic block coordinate descent for convex problems. In *Advances in Neural Information Processing Systems*, pages 1306–1314, 2015.
28. R. Sun and Y. Ye. Worst-case complexity of cyclic coordinate descent: $o(n^2)$ gap with randomized version. *arXiv preprint arXiv:1604.07130*, 2016.
29. R. S. Varga. *Matrix iterative analysis*. Springer Science & Business Media, 2009.
30. P.-W. Wang and C.-J. Lin. Iteration complexity of feasible descent methods for convex optimization. *Journal of Machine Learning Research*, 15:1523–1548, 2014.
31. S. J. Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.
32. S. J. Wright and C.-P. Lee. Analyzing Random Permutations for Cyclic Coordinate Descent. *ArXiv e-prints*, June 2017.
33. T. Zhang. A note on the non-commutative arithmetic-geometric mean inequality. *arXiv preprint arXiv:1411.5058*, November 2014.

A Proof of Lemma 2

Applying Lemma 1 with $Q = \Gamma^{-1}$, where Γ^{-1} is defined in (24), we get

$$\gamma = \frac{\sum_{j=0}^{n-2}(n-1-j)\alpha(1+\alpha)^j}{n(n-1)} = \frac{\alpha}{n}\sum_{j=0}^{n-2}(1+\alpha)^j - \frac{\alpha}{n(n-1)}\sum_{j=0}^{n-2} j(1+\alpha)^j$$
$$= \frac{(1+\alpha)^{n-1}-1}{n} - \frac{(1+\alpha)^{n-1}}{n} + \frac{(1+\alpha)^n - 1 - \alpha}{\alpha n(n-1)} = \frac{(1+\alpha)^n - \alpha n - 1}{\alpha n(n-1)},$$

where the third equality follows by the following lemma. This completes the proof.

Lemma 3 *For any real scalar $\eta \neq 1$ and integer $k \geq 0$, we have*

$$\sum_{j=0}^{k} j\eta^j = (k+1)\frac{\eta^{k+1}}{\eta-1} - \frac{(\eta^{k+1}-1)\eta}{(\eta-1)^2}.$$

Proof (Lemma 3) Consider the cumulative sums $u_k(\eta) := \sum_{j=0}^{k}\eta^j = \frac{\eta^{k+1}-1}{\eta-1}$. It is easy to see that $\sum_{j=0}^{k} j\eta^j = \eta u_k'(\eta)$ where $u_k'(\eta)$ is the derivative of $u_k(\eta)$. Differentiating the right-hand side of the formula for u_k yields the result.

B Proof of Proposition 1

Proof (Part (i)) : Defining $h(t,n) = \frac{\left(1+\frac{t}{n-1}\right)^n - 1}{\frac{t}{n-1}}$, where $t \in (0,1)$ and $n \geq 1$ is an integer, we have by the definition in (28) that $s(t,n) = \rho_1(t,n)/\rho_2(t,n)$, where

$$\rho_1(t,n) = -\log\left(1 - \frac{1-t}{n}h(t,n)\right) \quad \text{and} \quad \rho_2(t,n) = -n\log\left(1-\frac{1-t}{n}\right).$$

Throughout the rest of the proof, for simplicity, whenever the dependence of h, ρ_1 and ρ_2 on n is clear, we will abbreviate them by $h(t)$, $\rho_1(t)$ and $\rho_2(t)$, respectively. Similarly, whenever the dependence on t is also clear, we will abbreviate them by h, ρ_1 and ρ_2, respectively. In order to prove statement *(i)* of Proposition 1, it suffices to show that the partial derivative satisfies

$$\partial_t s(t,n) = \frac{\partial_t(\rho_1)\rho_2 - \rho_1\partial_t(\rho_2)}{\rho_2^2} < 0,$$

for all $t \in (0,1)$. This holds if and only if

$$\frac{\partial_t(\rho_1)}{\rho_1} < \frac{\partial_t(\rho_2)}{\rho_2} \quad \Longleftrightarrow \quad \partial_t\left(\log\rho_1\right) < \partial_t\left(\log\rho_2\right), \tag{35}$$

for all $t \in (0,1)$, where we used the fact that ρ_1 and ρ_2 are positive for $t \in (0,1)$. We can compute these partial derivatives in the right-hand side as follows

$$\partial_t \left(\log \rho_1\right) = \frac{1}{\rho_1}\partial_t(\rho_1) = \frac{-1}{\rho_1}\left(\frac{1}{1-\frac{1-t}{n}h(t)}\right)\left(\frac{h(t)+h'(t)(t-1)}{n}\right),$$

and similarly

$$\partial_t \left(\log \rho_2\right) = \frac{1}{\rho_2}\partial_t(\rho_2) = \frac{-1}{\rho_2}\left(\frac{1}{1-\frac{1-t}{n}}\right).$$

Hence, in order to prove (35), it is sufficient to show that

$$\frac{1}{\rho_1}\left(\frac{1}{1-\frac{1-t}{n}h(t)}\right)q(t) > \frac{1}{\rho_2}\left(\frac{1}{1-\frac{1-t}{n}}\right), \quad \text{where} \quad q(t) := \frac{h(t)+h'(t)(t-1)}{n},$$

which, after inserting the formulas for ρ_1 and ρ_2, is equivalent to

$$-n\log\left(1-\frac{1-t}{n}\right)\left(1-\frac{1-t}{n}\right)q(t) > -\log\left(1-\frac{1-t}{n}h(t)\right)\left(1-\frac{1-t}{n}h(t)\right), \tag{36}$$

for $t \in (0,1)$. The main ingredients to prove this inequality is to approximate the non-linear functions q and h with piecewise linear functions, which are easier to deal with, in other words, linearizing q and h above leads to simpler expressions for the derivatives of both sides of this inequality. In order to approximate q, we first write a binomial expansion for $h(t)$ as follows

$$h(t) = \frac{\left(1+\frac{t}{n-1}\right)^n - 1}{\frac{t}{n-1}} = \sum_{i=1}^{n}\binom{n}{i}\left(\frac{t}{n-1}\right)^{i-1}.$$

This implies that $q(t)$ is of the form $q(t) = \frac{1}{2} + \frac{2}{3}t + \sum_{j=2}^{n-1} c_j t^j$, where $c_2 > 0$ and $c_j \geq 0$, for all $j \in \{3,\ldots,n-1\}$. Therefore, the first and second derivatives of q are positive over $t \in (0,1)$ and q is strictly convex. We then consider linearizations of $q(t)$ at $t=0$ and $t=1$, which are given by

$$q_0(t) = \frac{1}{2} + \frac{2}{3}t \quad \text{and} \quad q_1(t) = \frac{h(1) - 2(n-1)(1-t)}{n}.$$

(Note that in the special case $n=2$, $q(t)$ is linear so that $q_0(t) = q_1(t)$ for all t. However, for $n > 2$, $q_0 \neq q_1$). In particular, it can be checked that $q_0(\hat{t}) = q_1(\hat{t})$, for $\hat{t} = 1 - \frac{6h(1)-7n}{4(2n-3)}$. Since $q(t)$ is convex,

$$q(t) \geq \underline{q}(t) = \max\left(q_0(t), q_1(t)\right) = \begin{cases} q_0(t), & \text{if } t \in [0,\hat{t}), \\ q_1(t), & \text{if } t \in [\hat{t},1]. \end{cases} \tag{37}$$

The right-hand side of (36) is of the form

$$z(t) = -\log\left(y(t)\right)y(t) = E(y(t)), \quad \text{where} \quad y(t) = 1 - \frac{1-t}{n}h(t), \quad E(y) = -\log(y)y. \tag{38}$$

As h is convex, we have the bounds

$$\overline{h}(t) = (1-t)h(0) + th(1) \geq h(t) \quad \text{and} \quad y(t) \geq \overline{y}(t) = 1 - \frac{1-t}{n}\overline{h}(t), \quad t \in (0,1). \tag{39}$$

Using the facts that the function $E(\cdot)$ has a maximum of $1/e$ over the interval $[0,1]$ and is strictly decreasing over the interval $(1/e,1]$, it follows from (39) that

$$E(y(t)) = z(t) \leq \overline{z}(t) := \begin{cases} E(\overline{y}(t)) & \text{if} \quad \overline{y} \in (1/e,1] \iff t \in (t_*,1] \\ 1/e & \text{if} \quad \overline{y} \in [0,1/e] \iff t \in [0,t_*] \end{cases} \tag{40}$$

where t_* is the largest $t \in (0,1)$ such that $\overline{y}(t) = 1/e$ and admits the formula

$$t_* = -\frac{1}{2}\frac{2n-h(1)}{h(1)-n} + \frac{1}{2}\sqrt{\left(\frac{2n-h(1)}{h(1)-n}\right)^2 + \frac{4}{e}\frac{n}{h(1)-n}}.$$

Combining the lower bound (37) on $q(t)$ and the upper bound (40) on $z(t)$, a sufficient condition for (36) is to show that the following relaxed inequality holds

$$-n\log\left(1-\frac{1-t}{n}\right)\left(1-\frac{1-t}{n}\right)\underline{q}(t) - \overline{z}(t) > 0, \quad \text{for all} \quad t \in (0,1). \tag{41}$$

The left-hand side is a piecewise continuously differentiable function (pieces defined by the intervals $[0,\hat{t}]$, $(\hat{t},t_*]$ and $(t_*,1]$)) and it is positive at $t=0$. The rest of the proof is about showing that the left-hand side in (41) stays positive for $t \in (0,1)$, this is achieved by computing and lower bounding the first order derivatives of the left-hand side. The details are skipped due to space considerations and follows from standard calculus techniques.

Proof (Part (ii)) : Since $\lim_{t\to 0^+}\rho_2(t) = -n\log(1-1/n)$, whereas $\lim_{t\to 0^+}\rho_1(t) = -\log(1-h(0)/n) = \infty$ as $h(0)=n$, we obtain $\lim_{t\to 0^+} s(t,n) = \lim_{t\to 0}(\rho_1(t)/\rho_2(t)) = \infty$.

Proof (Part (iii)) : We observe that $g(n) = \lim_{t\to 1^-}\frac{\rho_1(t)}{\rho_2(t)} = \lim_{t\to 1^-}\frac{\rho_1'(t)}{\rho_2'(t)}$, since $\lim_{t\to 1^-}\rho_1(t) = \lim_{t\to 1^-}\rho_2(t) = 0$. The derivatives of $\rho_1(t)$ and $\rho_2(t)$ with respect to t are given by

$$\rho_1'(t) = -\frac{h(t)+h'(t)(t-1)}{n-(1-t)h(t)} \quad \text{and} \quad \rho_2'(t) = -\frac{n}{n-(1-t)}.$$

Therefore, we obtain

$$g(n) = \lim_{t\to 1^-}\frac{\frac{h(t)+h'(t)(t-1)}{n-(1-t)h(t)}}{\frac{n}{n-(1-t)}} = \frac{h(1)}{n} = \left(1+\frac{1}{n-1}\right)^{n-1} + \frac{1}{n} - 1.$$

In order to show that $g(n)$ is strictly increasing in n, consider the extension of g to the positive real line, i.e., consider the function $\bar{g}(z) = \left(1+\frac{1}{z}\right)^z + \frac{1}{z+1} - 1$, where $z \geq 0$. Taking its derivative with respect to z, we get

$$\bar{g}'(z) = \left(\log\left(1+\frac{1}{z}\right) - \frac{1}{z+1}\right)\left(1+\frac{1}{z}\right)^z - \frac{1}{(z+1)^2}.$$

Using the lower bounds $\log(1+y) \geq \frac{2y}{2+y}$ for $y \geq 0$ and $(1+1/y)^y \geq 2$ for $y \geq 1$, we obtain

$$\bar{g}'(z) \geq 2\left(\frac{2}{2z+1} - \frac{1}{z+1}\right) - \frac{1}{(z+1)^2} = \frac{1}{(z+1)(z+1/2)} - \frac{1}{(z+1)^2} > 0,$$

for any $z \geq 1$. Consequently, $g(n)$ is strictly increasing in $n \geq 2$. Furthermore, it follows directly from the definition that $g(2) = 3/2$ and since $\lim_{n\to\infty}(1+1/n)^n = e$, we get $\lim_{n\to\infty} g(n) = e-1$. This completes the proof of part *(iii)*.

C Proof of Proposition 2

The proof of $\rho(B_{\mathrm{RPCD}}) < \rho(B_{\mathrm{RCD}})^n$ follows by Proposition 1, hence is omitted. Since the off-diagonal entries of A are nonpositive and A is a positive definite matrix, then it follows by [9, Theorem 4.12] that $\rho(B_{\mathrm{CCD}}) \leq \frac{1-\mu}{1+\mu} = 1 - \frac{2\mu}{1+\mu}$, where $\mu = 1-(n-1)\alpha$. On the other hand, from (27), we have $\rho(B_{\mathrm{RPCD}}) = 1 - \mu\frac{(1+\alpha)^n-1}{n\alpha}$. Hence, in order to show that $\rho(B_{\mathrm{CCD}}) < \rho(B_{\mathrm{RPCD}})$, for all $\alpha \in (1, 1/(n-1))$ and $n \geq 2$, it suffices to show

$$\frac{2}{1+\mu} > \frac{(1+\alpha)^n-1}{n\alpha} \quad \Longleftrightarrow \quad \frac{1}{1-\frac{(n-1)\alpha}{2}} > \frac{(1+\alpha)^n-1}{n\alpha}.$$

Since $\alpha \in (1, 1/(n-1))$, it is sufficient to show that

$$n\alpha > \left(1 - \frac{(n-1)\alpha}{2}\right)\left((1+\alpha)^n - 1\right). \tag{42}$$

Using the Binomial expansion $(1+\alpha)^n = \sum_{j=0}^{n} \binom{n}{j}\alpha^j$, we get

$$\begin{aligned}\left(1 - \frac{(n-1)\alpha}{2}\right)\left((1+\alpha)^n - 1\right) &= \sum_{j=1}^{n}\binom{n}{j}\alpha^j - \frac{n-1}{2}\sum_{j=1}^{n}\binom{n}{j}\alpha^{j+1} \\ &< \sum_{j=1}^{n}\binom{n}{j}\alpha^j - \frac{n-1}{2}\sum_{j=1}^{n-1}\binom{n}{j}\alpha^{j+1} \\ &= n\alpha + \sum_{j=2}^{n}\left(\binom{n}{j} - \frac{n-1}{2}\binom{n}{j-1}\right)\alpha^j,\end{aligned}$$

where the inequality follows since we omit the last term of the second sum and the last equality follows by peeling out the first entry of the first sum. We can observe that

$$\binom{n}{j} - \frac{n-1}{2}\binom{n}{j-1} = \left(\frac{n+1-j}{j} - \frac{n-1}{2}\right)\binom{n}{j-1} = \left(\frac{(n+1)(2-j)}{2j}\right)\binom{n}{j-1} \leq 0,$$

for all $j \in \{2, \ldots, n\}$. This proves (42), which concludes the proof.

D Proof of Proposition 3

RCD iterations can be written (by (8)) as follows

$$x_{\mathrm{RCD}}^{k+1} = \left(I - e_{i_k}e_{i_k}^T A\right)x_{\mathrm{RCD}}^k,$$

where i_k is drawn uniformly at random from the set $\{1, 2, \ldots, n\}$. Letting $\mathbb{E}_k$ denote the expectation with respect to i_k given x_k and taking norm squares of both sides, we obtain

$$\begin{aligned}\mathbb{E}_k\|x_{\mathrm{RCD}}^{k+1}\|^2 &= (x_{\mathrm{RCD}}^k)^T\,\mathbb{E}_k\left[\left(I - A^T e_{i_k}e_{i_k}^T\right)\left(I - e_{i_k}e_{i_k}^T A\right)\right]x_{\mathrm{RCD}}^k \\ &= (x_{\mathrm{RCD}}^k)^T\left(\frac{1}{n}\sum_{i=1}^{n}\left(I - A^T e_i e_i^T - e_i e_i^T A + A^T e_i e_i^T A\right)\right)x_{\mathrm{RCD}}^k \\ &= (x_{\mathrm{RCD}}^k)^T\left(I - \frac{2A}{n} + \frac{A^2}{n}\right)x_{\mathrm{RCD}}^k \leq \|Q\|\,\|x_{\mathrm{RCD}}^k\|^2 \text{ with } Q := I - \frac{2A}{n} + \frac{A^2}{n},\end{aligned}$$

where we used the fact that $A = A^T$ and $\sum_{i=1}^{n} e_i e_i^T = I$. Using this recursion and noting that $x^* = 0$, we get

$$\mathbb{E}\|x_{\mathrm{RCD}}^{k+1} - x^*\|^2 \leq \|Q\|^k\,\|x^0 - x^*\|^2. \tag{43}$$

The eigenvalues of Q are of the form $1 - 2\lambda/n + \lambda^2/n$, where λ is an eigenvalue of A. Since Q is symmetric and A has only two distinct eigenvalues that are equal to $\mu = (1-\alpha(n-1))$ and $L = 1+\alpha$, we obtain

$$\|Q\| = \max\{1 - 2\mu/n + \mu^2/n, 1 - 2L/n + L^2/n\} = 1 - 2\mu/n + \mu^2/n. \tag{44}$$

Using (44) in (43) concludes the proof of (31). The proof of (32) can be done by following similar lines to the above proof as follows

$$\begin{aligned}f(x_{\mathrm{RCD}}^{k+1}) &= (x_{\mathrm{RCD}}^k)^T\,\mathbb{E}_k\left[\left(I - A^T e_{i_k}e_{i_k}^T\right)A\left(I - e_{i_k}e_{i_k}^T A\right)\right]x_{\mathrm{RCD}}^k \\ &= (x_{\mathrm{RCD}}^k)^T\,\mathbb{E}_k\left[A - A^T e_{i_k}e_{i_k}^T A - A e_{i_k}e_{i_k}^T A + A^T e_{i_k}e_{i_k}^T A e_{i_k}e_{i_k}^T A\right]x_{\mathrm{RCD}}^k \\ &= (x_{\mathrm{RCD}}^k)^T\,\mathbb{E}_k\left[A - A e_{i_k}e_{i_k}^T A\right]x_{\mathrm{RCD}}^k \\ &= (x_{\mathrm{RCD}}^k)^T\left(A - \frac{A^2}{n}\right)x_{\mathrm{RCD}}^k \leq \left\|I - \frac{A}{n}\right\| f(x_{\mathrm{RCD}}^k) = \left(1 - \frac{\mu}{n}\right)f(x_{\mathrm{RCD}}^k),\end{aligned}$$

where in the third equality, we use the fact that $A = A^T$ and $e_i^T A e_i = 1$, for all $i \in [n]$, and in the fourth equality, we use $\sum_{i=1}^n e_i e_i^T = I$, respectively. This concludes the proof.

E Proof of Proposition 4

RPCD iterations can be written (by (10)) as follows

$$x_{\text{RPCD}}^{(\ell+1)n} = P_{\pi_\ell} B_{\text{CCD}} P_{\pi_\ell}^T x_{\text{RPCD}}^{\ell n}.$$

Considering improvement sequence $\mathcal{I}_2$, this yields

$$\mathbb{E}_\ell \|x_{\text{RPCD}}^{(\ell+1)n}\|^2 = (x_{\text{RPCD}}^{\ell n})^T \mathbb{E}_P[P B_{\text{CCD}}^T B_{\text{CCD}} P^T] x_{\text{RPCD}}^{\ell n} \leq \|S\| \|x_{\text{RPCD}}^{\ell n}\|^2,$$

where $S = \mathbb{E}_P[P B_{\text{CCD}}^T B_{\text{CCD}} P^T]$. Using this recursion, we obtain

$$\mathbb{E}\|x_{\text{RPCD}}^{\ell n}\|^2 \leq \|S\|^\ell \|x_{\text{RPCD}}^0\|^2.$$

The contraction factor $\|S\|$ can be computed by applying Lemma 1 with $Q = B_{\text{CCD}}^T B_{\text{CCD}}$, which yields

$$S = \mathbb{E}_P[P B_{\text{CCD}}^T B_{\text{CCD}} P^T] = \tau_1 I + \tau_2 \mathbf{1}\mathbf{1}^T, \tag{45}$$

where

$$\tau_2 = \frac{\mathbf{1}^T B_{\text{CCD}}^T B_{\text{CCD}} \mathbf{1} - \text{trace}(B_{\text{CCD}}^T B_{\text{CCD}})}{n(n-1)} \quad \text{and} \quad \tau_1 = \frac{\text{trace}(B_{\text{CCD}}^T B_{\text{CCD}})}{n} - \tau_2.$$

Since S is a symmetric matrix, we have $\|S\| = \rho(S)$. Furthermore, we can observe that $B_{\text{CCD}}^T B_{\text{CCD}}$ has strictly positive entries both in its diagonals and off-diagonals, consequently we have $S > 0$. Then, by Perron-Frobenius Theorem [29, Lemma 2.8], we have

$$\|S\| = \rho(S) = \tau_1 + n\tau_2 = \frac{1}{n} \mathbf{1}^T S \mathbf{1}. \tag{46}$$

In order to compute (46), we first compute the matrix B_{CCD} as follows

$$B_{\text{CCD}} = I - \Gamma^{-1} A = \begin{cases} \alpha\left((1+\alpha)^{i-1} - (1+\alpha)^{i-j}\right), & \text{if} \quad i \geq j, \\ \alpha(1+\alpha)^{i-1}, & \text{if} \quad i < j. \end{cases} \tag{47}$$

Combining (46) and (47), we obtain

$$\|S\| = \frac{1}{n} \mathbf{1}^T B_{\text{CCD}}^T B_{\text{CCD}} \mathbf{1} = \frac{1}{n} \|B_{\text{CCD}} \mathbf{1}\|^2 = \frac{1}{n} \sum_{i=1}^n \left((B_{\text{CCD}} \mathbf{1})_i\right)^2,$$

where

$$(B_{\text{CCD}} \mathbf{1})_i = 1 - \mu(1+\alpha)^{i-1}. \tag{48}$$

This yields

$$\|S\| = \frac{1}{n} \sum_{i=1}^n \left(1 - 2\mu(1+\alpha)^{i-1} + \mu^2 (1+\alpha)^{2(i-1)}\right) = 1 - \frac{2\mu}{n}\left(\frac{(1+\alpha)^n - 1}{\alpha}\right) + \frac{\mu^2}{n}\left(\frac{(1+\alpha)^{2n} - 1}{\alpha(\alpha+2)}\right),$$

which proves (33).

We next prove the results regarding the function suboptimality in (34). To this end, we consider the expected function sub-optimality (note that $f(x^*) = 0$), which yields

$$\begin{aligned}
\mathbb{E}_\ell f(x_{\text{RPCD}}^{(\ell+1)n}) &= (x_{\text{RPCD}}^{\ell n})^T \mathbb{E}_P[P B_{\text{CCD}}^T P^T A P B_{\text{CCD}} P^T] x_{\text{RPCD}}^{\ell n} \\
&= (x_{\text{RPCD}}^{\ell n})^T \mathbb{E}_P[P B_{\text{CCD}}^T A B_{\text{CCD}} P^T] x_{\text{RPCD}}^{\ell n} \\
&\leq \|\mathbb{E}_P[A^{-1/2} P B_{\text{CCD}}^T A B_{\text{CCD}} P^T A^{-1/2}]\| \|A^{1/2} x_{\text{RPCD}}^{\ell n}\|^2 \\
&= \|\mathbb{E}_P[A^{-1/2} P B_{\text{CCD}}^T A B_{\text{CCD}} P^T A^{-1/2}]\| \, f(x_{\text{RPCD}}^{\ell n}) \\
&= \|\mathbb{E}_P[P A^{-1/2} B_{\text{CCD}}^T A B_{\text{CCD}} A^{-1/2} P^T]\| \, f(x_{\text{RPCD}}^{\ell n}) \\
&= \|G\| \, f(x_{\text{RPCD}}^{\ell n}),
\end{aligned}$$

where $G := \mathbb{E}_P[PA^{-1/2}B_{\rm CCD}^T AB_{\rm CCD}A^{-1/2}P^T]$ and the equalities follow since A and $A^{-1/2}$ are symmetric permutation invariant matrices, i.e., $PAP^T = A$ and $PA^{-1/2}P^T = A^{-1/2}$. It can be shown that $A^{1/2}B_{\rm CCD}A^{-1/2}$ is a non-negative matrix, hence applying Lemma 1 to the matrix $Q = A^{-1/2}B_{\rm CCD}^T AB_{\rm CCD}A^{-1/2}$, it can be shown (similar to the previous proof) that

$$\|G\| = \rho(G) = \frac{1}{n}\|A^{1/2}B_{\rm CCD}A^{-1/2}\mathbf{1}\|^2 = \frac{1}{n}\|\mathbf{1} - A^{1/2}\Gamma^{-1}A^{1/2}\mathbf{1}\|^2, \tag{49}$$

where $A^{1/2} = \gamma I - \sigma \mathbf{1}\mathbf{1}^T$ with $\gamma = \sqrt{1+\alpha}$ and $\sigma = (\gamma - \sqrt{\mu})/n$. This yields $A^{1/2}\mathbf{1} = (\gamma - n\sigma)\mathbf{1} = \sqrt{\mu}\mathbf{1}$. Multiplying both sides of the above equality by Γ^{-1} from the left, we obtain

$$\Gamma^{-1}A^{1/2}\mathbf{1} = \sqrt{\mu}\, c, \tag{50}$$

where it follows from (24) that

$$c = \begin{bmatrix} 1 \\ 1+\alpha \\ 1+\alpha+\alpha(1+\alpha) \\ \vdots \\ 1+\alpha+\alpha(1+\alpha)+\cdots+\alpha(1+\alpha)^{n-2} \end{bmatrix} = \begin{bmatrix} 1 \\ 1+\alpha \\ (1+\alpha)^2 \\ \vdots \\ (1+\alpha)^{n-1} \end{bmatrix}.$$

Multiplying (50) from the left by $A^{1/2}$, we get

$$A^{1/2}\Gamma^{-1}A^{1/2}\mathbf{1} = \sqrt{\mu}\left(\gamma c - \sigma\, ||c||_1\, \mathbf{1}\right), \quad \text{where} \quad ||c||_1 = \frac{(1+\alpha)^n - 1}{\alpha}. \tag{51}$$

Using (51) in (49), we obtain

$$\begin{aligned} ||G|| &= \frac{1}{n}\sum_{i=1}^n \left(1 - \sqrt{\mu}\left(\gamma c_i - \sigma\, ||c||_1\right)\right)^2 = 1 - \frac{2\sqrt{\mu}}{n}\sum_{i=1}^n \left(\gamma c_i - \sigma\, ||c||_1\right) + \frac{\mu}{n}\sum_{i=1}^n \left(\gamma c_i - \sigma\, ||c||_1\right)^2 \\ &= 1 - \frac{2\sqrt{\mu}}{n}\left(\gamma - n\sigma\right)||c||_1 + \frac{\mu}{n}\sum_{i=1}^n \left(\gamma^2 c_i^2 - 2\gamma\sigma\, ||c||_1\, c_i + \sigma^2\, ||c||_1^2\right) \\ &= 1 - \frac{2\mu}{n}||c||_1 + \frac{\mu}{n}\left(\gamma^2\, ||c||_2^2 - 2\gamma\sigma\, ||c||_1^2 + n\sigma^2\, ||c||_1^2\right), \end{aligned} \tag{52}$$

where

$$||c||_2^2 = \frac{(1+\alpha)^{2n} - 1}{\alpha(\alpha+2)} \quad \text{and} \quad ||c||_1^2 = \frac{(1+\alpha)^{2n} - 2(1+\alpha)^n + 1}{\alpha^2}.$$

Modifying the terms in (52), we get

$$\begin{aligned} ||G|| &= 1 - \frac{2\mu}{n}||c||_1 + \frac{\mu}{n}\left(\gamma^2\, ||c||_2^2 - \gamma\sigma\, ||c||_1^2 + \sigma(n\sigma - \gamma)\, ||c||_1^2\right) \\ &= 1 - \frac{2\mu}{n}||c||_1 + \frac{\mu}{n}\left((1+\alpha)\, ||c||_2^2 - \frac{1+\alpha-(1-\alpha(n-1))}{n}\, ||c||_1^2\right) \\ &= 1 - \frac{2\mu}{n}||c||_1 + \frac{\mu}{n}\left((1+\alpha)\, ||c||_2^2 - \alpha\, ||c||_1^2\right) \\ &= 1 - \frac{2\mu}{n}||c||_1 + \frac{\mu}{n}\left((1+\alpha)\frac{(1+\alpha)^{2n}-1}{\alpha(\alpha+2)} - \frac{(1+\alpha)^{2n} - 2(1+\alpha)^n + 1}{\alpha}\right) \\ &= 1 - \frac{\mu}{n}\left(\frac{(1+\alpha)^{2n}-1}{\alpha(\alpha+2)}\right), \end{aligned}$$

which concludes the proof of Proposition 4.