CORE DISCUSSION PAPER

2010/2

Efficiency of coordinate descent methods on huge-scale optimization problems

Yu. Nesterov *

January 2010

Abstract

In this paper we propose new methods for solving huge-scale optimization problems. For problems of this size, even the simplest full-dimensional vector operations are very expensive. Hence, we propose to apply an optimization technique based on random partial update of decision variables. For these methods, we prove the global estimates for the rate of convergence. Surprisingly enough, for certain classes of objective functions, our results are better than the standard worst-case bounds for deterministic algorithms. We present constrained and unconstrained versions of the method, and its accelerated variant. Our numerical test confirms a high efficiency of this technique on problems of very big size.

Keywords: Convex optimization, coordinate relaxation, worst-case efficiency estimates, fast gradient schemes, Google problem.

*Center for Operations Research and Econometrics (CORE), Université catholique de Louvain (UCL), 34 voie du Roman Pays, B-1348 Louvain-la-Neuve, Belgium; e-mail: Yurii.Nesterov@uclouvain.be.

The research results presented in this paper have been supported by a grant "Action de recherche concertè ARC 04/09-315" from the "Direction de la recherche scientifique - Communautè française de Belgique". The scientific responsibility rests with its author(s).

1 Introduction

Motivation. Coordinate descent methods were among the first optimization schemes suggested for solving smooth unconstrained minimization problems (see [1, 2] and references therein). The main advantage of these methods is the simplicity of each iteration, both in generating the search direction and in performing the update of variables. However, very soon it became clear that the coordinate descent methods can be criticized in several aspects.

1. Theoretical justification. The simplest variant of the coordinate descent method is based on a cyclic coordinate search. However, for this strategy it is difficult to prove convergence, and almost impossible to estimate the rate of convergence[1)]. Another possibility is to move along the direction corresponding to the component of gradient with maximal absolute value. For this strategy, justification of the convergence rate is trivial. For the future references, let us present this result right now. Consider the unconstrained minimization problem

$$\min_{x \in R^n} f(x), \tag{1.1}$$

where the convex objective function f has component-wise Lipschitz continuous gradient:

$$|\nabla_i f(x + he_i) - \nabla_i f(x)| \leq M|h|, \quad x \in R^n,\ h \in R,\ i = 1, \ldots, n, \tag{1.2}$$

where e_i is the ith coordinate vector in R^n. Consider the following method:

Choose $x_0 \in R^n$. For $k \geq 0$ iterate

1. Choose $i_k = \arg \max\limits_{1 \leq i \leq n} |\nabla_i f(x_k)|$.
2. Update $x_{k+1} = x_k - \frac{1}{M} \nabla_{i_k} f(x_k) e_{i_k}$.

(1.3)

Then

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\overset{(1.2)}{\geq} \frac{1}{2M} |\nabla_{i_k} f(x_k)|^2 \geq \frac{1}{2nM} \|\nabla f(x_k)\|^2 \\ &\geq \frac{1}{2nMR^2} (f(x_k) - f^*)^2, \end{aligned}$$

where $R \geq \|x_0 - x^*\|$, the norm is Euclidean, and f^* is the optimal value of problem (1.1). Hence,

$$f(x_k) - f^* \leq \frac{2nMR^2}{k+4}, \quad k \geq 0. \tag{1.4}$$

Note that at each test point, method (1.3) requires computation of the *whole* gradient vector. However, if this vector is available, it seems better to apply the usual full-gradient methods. It is also important that for convex functions with Lipschitz-continuous gradient:

$$\|\nabla f(x) - \nabla f(y)\| \leq L(f)\|x - y\|, \quad x, y \in R^n, \tag{1.5}$$

[1)] To the best of our knowledge, in general case this is not done up to now.

it can happen that $M \geq O(L(f))$. Hence, in general, the rate (1.4) is *worse* than the rate of convergence of the simple Gradient Method (e.g. Section 2.1 in [6]).

2. Computational complexity. From the theory of fast differentiation it is known that for all functions defined by an explicit sequence of standard operations, the complexity of computing the whole gradient is proportional to the computational complexity of the value of corresponding function. Moreover, the coefficient in this proportion is a small absolute constant. This observation suggests that for coordinate descent methods the line-search strategies based on the function values are too expensive. Provided that the general directional derivative of the function has the same complexity as the function value, it seems that no room is left for supporting the coordinate descent idea. The versions of this method still appear in the literature. However, they are justified only by local convergence analysis for rather particular problems (e.g. [3, 8]).

At the same time, in the last years we can observe an increasing interest to optimization problems of a very big size (Internet applications, telecommunications). In such problems, even computation of a function value can require substantial computational efforts. Moreover, some parts of the problem's data can be distributed in space and in time. The problem's data may be only partially available at the moment of evaluating the current test point. For problems of this type, we adopt the term *huge-scale* problems.

These applications strongly push us backward to the framework of coordinate minimization. Therefore, let us look again at the above criticism. It appears, that there is a small chance for these methods to survive.

1. We can also think about the *random* coordinate search with pre-specified probabilities for each coordinate move. As we will see later, the complexity analysis of corresponding methods is quite straightforward. On the other hand, from technological point of view, this strategy fits very well the problems with distributed or unstable data.

2. It appears, that the computation of a *coordinate directional derivative* can be much simpler than computation of either a function value, or a directional derivative along *arbitrary* direction.

In order to support this claim, let us look at the following optimization problem:

$$\min_{x \in R^n} \left\{ f(x) \stackrel{\text{def}}{=} \sum_{i=1}^{n} f_i(x^{(i)}) + \tfrac{1}{2}\|Ax - b\|^2 \right\}, \tag{1.6}$$

where f_i are convex differentiable univariate functions, $A = (a_1, \ldots, a_n) \in R^{p \times n}$ is an $p \times n$-matrix, and $\|\cdot\|$ is the standard Euclidean norm in R^p. Then

$$\begin{aligned} \nabla_i f(x) &= f_i'(x^{(i)}) + \langle a_i, g(x) \rangle, \quad i = 1, \ldots, n, \\ g(x) &= Ax - b. \end{aligned}$$

If the residual vector $g(x)$ is already computed, then the computation of ith directional derivative requires $O(p_i)$ operations, where p_i is the number of nonzero elements in vector a_i. On the other hand, the coordinate move $x_+ = x + \alpha e_i$ results in the following change in the residual:

$$g(x_+) = g(x) + \alpha a_i.$$

Therefore, the ith coordinate step for problem (1.6) needs $O(p_i)$ operations. Note that computation of either the function value, or the whole gradient, or an arbitrary directional derivative requires $O\left(\sum\limits_{i=1}^{n} p_i\right)$ operations. The reader can easily find many other examples of optimization problems with cheap coordinate directional derivatives.

The goal of this paper is to provide the random coordinate descent methods with the worst-case efficiency estimates. We show that for functions with cheap coordinate derivatives the new methods are always faster than the corresponding full-gradient schemes. A surprising result is that even for functions with expensive coordinate derivatives the new methods can be also faster since their rate of convergence depends on an upper estimate for the *average* diagonal element of the Hessian of the objective function. This value can be much smaller than the maximal eigenvalue of the Hessian, entering the worst-case complexity estimates of the black-box gradient schemes.

Contents. The paper is organized as follows. In Section 2 we analyze the expected rate of convergence of the simplest Random Coordinate Descent Method. It is shown that it is reasonable to define probabilities for particular coordinate directions using the estimates of Lipschitz constants for partial derivatives. In Section 3 we show that for the class of strongly convex functions RCDM converges with a linear rate. By applying a regularization technique, this allows to solve the unconstrained minimization problems with arbitrary high confidence level. In Section 4 we analyze a modified version of RCDM as applied to constrained optimization problem. In Section 5 we show that unconstrained version of RCDM can be accelerated up to the rate of convergence $O(\frac{1}{k^2})$, where k is the iteration counter. Finally, in Section 6 we discuss the implementation issues. In Section 6.1 we show that the good estimates for coordinate Lipshitz constants can be efficiently computed. In Section 6.2 we present an efficient strategy for generating random coordinate directions. And in Section 6.3 we present preliminary computational results.

Notation. In this paper we work with real coordinate spaces R^n composed by column vectors. For $x, y \in R^n$ denote

$$\langle x, y \rangle \;=\; \sum_{i=1}^{n} x^{(i)} y^{(i)}.$$

We use the same notation $\langle \cdot, \cdot \rangle$ for spaces of different dimension. Thus, its actual sense is defined by the space containing the arguments. If we fix a norm $\|\cdot\|$ in R^n, then the dual norm is defined in the standard way:

$$\|s\|^* \;=\; \max_{\|h\|=1} \langle s, x \rangle. \tag{1.7}$$

We denote by $s^\#$ an arbitrary vector from the set

$$\operatorname{Arg}\max_x \left[\langle s, x \rangle - \tfrac{1}{2}\|x\|^2 \right]. \tag{1.8}$$

Clearly, $\|s^\#\| = \|s\|^*$.

For function $f(x)$, $x \in R^n$, we denote by $\nabla f(x)$ its *gradient*, which is a vector from R^n composed by partial derivatives. By $\nabla^2 f(x)$ we denote the Hessian of f at x. In the sequel we will use the following simple result.

Lemma 1 *Let us fix a decomposition of R^n on k subspaces. For positive semidefinite symmetric matrix $A \in R^{n\times n}$, denote by $A_{i,i}$ the corresponding diagonal blocks. If*

$$A_{i,i} \preceq L_i B_i, \quad i = 1, \ldots, k,$$

where all $B_i \succeq 0$, then

$$A \preceq \left(\sum_{i=1}^{k} L_i \right) \cdot \mathrm{diag}\, \{B_i\}_{i=1}^{k},$$

where $\mathrm{diag}\, \{B_i\}_{i=1}^{k}$ *is a block-diagonal $(n \times N)$-matrix with diagonal blocks B_i. In particular,*

$$A \preceq n \cdot \mathrm{diag}\, \{A_{i,i}\}_{i=1}^{n}.$$

Proof:
Denote by $x^{(i)}$ the part of vector $x \in R^n$, belonging to ith subspace. Since A is positive semidefinite, we have

$$\begin{aligned} \langle Ax, x \rangle &= \sum_{i=1}^{k} \sum_{j=1}^{k} \langle A_{i,j} x^{(i)}, x^{(j)} \rangle \le \left(\sum_{i=1}^{k} \langle A_{i,i} x^{(i)}, x^{(i)} \rangle^{1/2} \right)^2 \\ &\le \left(\sum_{i=1}^{k} L_i^{1/2} \langle B_{i,i} x^{(i)}, x^{(i)} \rangle^{1/2} \right)^2 \le \sum_{i=1}^{k} L_i \cdot \sum_{i=1}^{k} \langle B_i x^{(i)}, x^{(i)} \rangle. \end{aligned}$$

□

2 Coordinate relaxation for unconstrained minimization

Consider the following unconstrained minimization problem

$$\min_{x \in R^N} f(x), \tag{2.1}$$

where the objective function f is convex and differentiable on R^N. We assume that the optimal set X^* of this problem is nonempty and bounded.

Let us fix a decomposition of R^N on n subspaces:

$$R^N = \bigotimes_{i=1}^{n} R^{n_i}, \quad N = \sum_{i=1}^{n} n_i.$$

Then we can define the corresponding partition of the unit matrix

$$I_N = (U_1, \ldots, U_n) \in R^{N\times N}, \quad U_i \in R^{N \times n_i},\ i = 1, \ldots, n.$$

Thus, any $x = (x^{(1)}, \ldots, x^{(n)})^T \in R^N$ can be represented as

$$x = \sum_{i=1}^{n} U_i x^{(i)}, \quad x^{(i)} \in R^{n_i},\ i = 1, \ldots, n.$$

Then the *partial gradient* of $f(x)$ in $x^{(i)}$ is defined as

$$f_i'(x) = U_i^T \nabla f(x) \in R^{n_i}, \quad x \in R^N.$$

For spaces R^{n_i}, let us fix some norms $\|\cdot\|_{(i)}$, $i = 1, \dots, n$. We assume that the gradient of function f is coordinate-wise Lipschitz continuous with constants $L_i = L_i(f)$:

$$\|f_i'(x + U_i h_i) - f_i'(x)\|_{(i)}^* \;\leq\; L_i \|h_i\|_{(i)}, \quad h_i \in R^{n_i},\ i = 1, \dots, n,\ x \in R^N. \tag{2.2}$$

For the sake of simplicity, let us assume that these constants are known.

By the standard reasoning (e.g. Section 2.1 in [6]), we can prove that

$$f(x + U_i h_i) \leq f(x) + \langle f_i'(x), h_i \rangle + \tfrac{L_i}{2} \|h_i\|_{(i)}^2, \quad x \in R^N,\ h_i \in R^{n_i},\ i = 1, \dots, n. \tag{2.3}$$

Let us define the optimal coordinate steps:

$$T_i(x) \;\stackrel{\text{def}}{=}\; x - \tfrac{1}{L_i} U_i f_i'(x)^{\#}, \quad i = 1, \dots, n,$$

Then, in view of the bound (2.3), we get

$$f(x) - f(T_i(x)) \;\geq\; \tfrac{1}{2L_i} \left(\|f_i'(x)\|_{(i)}^* \right)^2, \quad i = 1, \dots, n. \tag{2.4}$$

In our random algorithm, we need a special random counter $\mathcal{R}_\alpha$, $\alpha \in R$, which generates an integer number $i \in \{1, \dots, n\}$ with probability

$$p_\alpha^{(i)} \;=\; L_i^\alpha \cdot \left[\sum_{j=1}^n L_j^\alpha \right]^{-1}, \quad i = 1, \dots, n. \tag{2.5}$$

Thus, the operation $i = \mathcal{R}_\alpha$ means that an integer random value, chosen from the set $\{1, \dots, n\}$ in accordance to probabilities (2.5), is assigned to variable i. Note that $\mathcal{R}_0$ generates a uniform distribution.

Now we can present the scheme of Random Coordinate Descent Method. It needs a starting point $x_0 \in R^N$ and value $\alpha \in R$ as input parameters.

Method $RCDM(\alpha, x_0)$

For $k \geq 0$ iterate:

1) Choose $i_k = \mathcal{R}_\alpha$.

2) Update $x_{k+1} = T_{i_k}(x_k)$. (2.6)

For estimating the rate of convergence of $RCDM$, we introduce the following norms:

$$\begin{aligned} \|x\|_\alpha &= \left[\sum_{i=1}^n L_i^\alpha \|x^{(i)}\|_{(i)}^2 \right]^{1/2}, \ x \in R^N, \\ \|g\|_\alpha^* &= \left[\sum_{i=1}^n L_i^{-\alpha} \left(\|g^{(i)}\|_{(i)}^* \right)^2 \right]^{1/2}, \ g \in R^N. \end{aligned} \tag{2.7}$$

Clearly, these norms satisfy Cauchy-Schwartz inequality:

$$\|g\|_\alpha^* \cdot \|x\|_\alpha \;\geq\; \langle g, x\rangle, \quad x, g \in R^N. \tag{2.8}$$

In the sequel, we use notation $S_\alpha = S_\alpha(f) = \sum\limits_{i=1}^n L_i^\alpha(f)$ with $\alpha \geq 0$. Note that

$$S_0(f) \;\equiv\; n.$$

Let us link our main assumption (2.2) with a full-dimensional Lipschitz condition for the gradient of the objective function.

Lemma 2 *Let f satisfy condition (2.2). Then for any $\alpha \in R$ we have*

$$\|\nabla f(x) - \nabla f(y)\|_{1-\alpha}^* \;\leq\; S_\alpha \|x - y\|_{1-\alpha}, \quad x, y \in R^N. \tag{2.9}$$

Therefore,

$$f(x) \;\leq\; f(y) + \langle \nabla f(y), x - y\rangle + \tfrac{1}{2} S_\alpha \|x - y\|_{1-\alpha}^2, \quad x, y \in R^N. \tag{2.10}$$

Proof:
Indeed,

$$\begin{aligned} f(x) - f^* &\overset{(2.4)}{\geq} \max_{1 \leq i \leq n} \tfrac{1}{2L_i} \left(\|f_i'(x)\|_{(i)}^* \right)^2 \\ &\geq \tfrac{1}{2S_\alpha} \sum_{i=1}^n \tfrac{L_i^\alpha}{L_i} \left(\|f_i'(x)\|_{(i)}^* \right)^2 \;=\; \tfrac{1}{2S_\alpha} \left(\|\nabla f(x)\|_{1-\alpha}^* \right)^2. \end{aligned}$$

Applying this inequality to function $\phi(x) = f(x) - f(y) - \langle \nabla f(y), x - y\rangle$, we obtain

$$f(x) \;\geq\; f(y) + \langle \nabla f(y), x - y\rangle + \tfrac{1}{2S_\alpha} \left(\|\nabla f(x) - \nabla f(y)\|_{1-\alpha}^* \right)^2, \quad x, y \in R^N. \tag{2.11}$$

Adding two variants of (2.11) with x an y interchanged, we get

$$\begin{aligned} \tfrac{1}{S_\alpha} \left(\|\nabla f(x) - \nabla f(y)\|_{1-\alpha}^* \right)^2 &\leq \langle \nabla f(x) - \nabla f(y), x - y\rangle \\ &\overset{(2.8)}{\leq} \|\nabla f(x) - \nabla f(y)\|_{1-\alpha}^* \cdot \|x - y\|_{1-\alpha}. \end{aligned}$$

Inequality (2.10) can be derived from (2.9) by simple integration. □

After k iterations, $RCDM(\alpha, x_0)$ generates a random output $(x_k, f(x_k))$, which depends on the observed implementation of random variable

$$\xi_k \;=\; \{i_0, \ldots, i_k\}.$$

Let us show that the expected value

$$\phi_k \;\overset{\text{def}}{=}\; E_{\xi_{k-1}} f(x_k)$$

converges to the optimal value f^* of problem (2.1).

Theorem 1 *For any $k \geq 0$ we have*

$$\phi_k - f^* \;\leq\; \tfrac{2}{k+4} \cdot \left[\sum_{j=1}^{n} L_j^{\alpha} \right] \cdot R_{1-\alpha}^2(x_0), \tag{2.12}$$

where $R_\beta(x_0) = \max\limits_x \left\{ \max\limits_{x_* \in X^*} \|x - x_*\|_\beta : \; f(x) \leq f(x_0) \right\}$.

Proof:
Let $RCDM$ generate implementation x_k of corresponding random variable. Then

$$\begin{aligned} f(x_k) - E_{i_k}(f(x_{k+1})) \;&=\; \sum_{i=1}^{n} p_\alpha^{(i)} \cdot [f(x_k) - f(T_i(x_k))] \\ &\overset{(2.4)}{\geq} \sum_{i=1}^{n} \tfrac{p_\alpha^{(i)}}{2L_i} \left(\|f_i'(x_k)\|_{(i)}^* \right)^2 \overset{(2.5)}{=} \tfrac{1}{2S_\alpha} (\|\nabla f(x_k)\|_{1-\alpha}^*)^2. \end{aligned} \tag{2.13}$$

Note that $f(x_k) \leq f(x_0)$. Therefore,

$$f(x_k) - f^* \;\leq\; \min_{x_* \in X^*} \langle \nabla f(x_k), x_k - x_* \rangle \overset{(2.8)}{\leq} \|\nabla f(x_k)\|_{1-\alpha}^* R_{1-\alpha}(x_0).$$

Hence,

$$f(x_k) - E_{i_k}(f(x_{k+1})) \;\geq\; \tfrac{1}{C}(f(x_k) - f^*)^2,$$

where $C \overset{\text{def}}{=} 2S_\alpha R_{1-\alpha}^2(x_0)$. Taking the expectation of both sides of this inequality in ξ_{k-1}, we obtain

$$\phi_k - \phi_{k+1} \;\geq\; \tfrac{1}{C} E_{\xi_{k-1}}[(f(x_k) - f^*)^2] \;\geq\; \tfrac{1}{C}(\phi_k - f^*)^2.$$

Thus,

$$\tfrac{1}{\phi_{k+1} - f^*} - \tfrac{1}{\phi_k - f^*} \;=\; \tfrac{\phi_k - \phi_{k+1}}{(\phi_{k+1} - f^*)(\phi_k - f^*)} \;\geq\; \tfrac{\phi_k - \phi_{k+1}}{(\phi_k - f^*)^2} \;\geq\; \tfrac{1}{C},$$

and we conclude that $\tfrac{1}{\phi_k - f^*} \geq \tfrac{1}{\phi_0 - f^*} + \tfrac{k}{C} \overset{(2.10)}{\geq} \tfrac{k+4}{C}$ for any $k \geq 0$. □

Let us look at the most important variants of the estimate (2.12).

- $\alpha = 0$. Then $S_0 = n$, and we get

$$\phi_k - f^* \;\leq\; \tfrac{2n}{k+4} \cdot R_1^2(x_0). \tag{2.14}$$

  Note that problem (2.1) can be solved by the standard full-gradient method endowed with the metric $\|\cdot\|_1$. Then its rate of convergence can be estimated as

$$f(x_k) - f^* \;\leq\; \tfrac{\gamma}{k} R_1^2(x_0),$$

  where the constant γ is big enough to ensure

$$f''(x) \preceq \gamma \cdot \operatorname{diag} \{L_i \cdot I_{n_i}\}_{i=1}^n, \quad x \in E,$$

  and I_k is a unit matrix in R^k. (Assume for a moment that f is twice differentiable.) However, since the constants L_i are the upper bounds for the block-diagonal elements of the Hessian, in the worst case we have $\gamma = n$. Hence, the worst-case rate of convergence of this variant of the gradient method is proportional to that one of $RCDM$. However, the iteration of the latter method is usually much cheaper.

- $\alpha = \frac{1}{2}$. Consider the case $n_i = 1$, $i = 1, \dots, n$. Denote by $D_\infty(x_0)$ the size of the initial level set measured in the infinity-norm:

$$D_\infty(x_0) = \max_x \left\{ \max_{y \in X^*} \max_{1 \le i \le n} |x^{(i)} - y^{(i)}| : \; f(x) \le f(x_0) \right\}.$$

  Then, $R^2_{1/2}(x_0) \le S_{1/2} D^2_\infty(x_0)$, and we obtain

$$\phi_k - f^* \;\; \le \;\; \frac{2}{k+4} \cdot \left[\sum_{i=1}^n L_i^{1/2} \right]^2 \cdot D^2_\infty(x_0). \tag{2.15}$$

  Note that for the first order methods, the worst-case dimension-independent complexity of minimizing the convex functions over an n-dimensional box is infinite [4]. Since for some problems the value $S_{1/2}$ can be bounded even for very big (or even infinite) dimension, the estimate (2.15) shows that RCDM can work in situations where the usual gradient methods have no theoretical justification.

- $\alpha = 1$. Consider the case when all norms $\| \cdot \|_{(i)}$ are the standard Euclidean norms of R^{n_i}, $i = 1, \dots, n$. Then $R_0(x_0)$ is the size of the initial level set in the standard Euclidean norm of R^N, and the rate of convergence of $RCDM(1, x_0)$ is as follows:

$$\phi_k - f^* \;\; \le \;\; \frac{2}{k+4} \cdot \left[\sum_{i=1}^n L_i \right] \cdot R_0^2(x_0) \;\; \equiv \;\; \frac{2n}{k+4} \cdot \left[\frac{1}{n} \sum_{i=1}^n L_i \right] \cdot R_0^2(x_0). \tag{2.16}$$

  At the same time, the rate of convergence of the standard gradient method can be estimated as

$$f(x_k) - f^* \le \frac{\gamma}{k} R_0^2(x_0),$$

  where γ satisfies condition

$$f''(x) \preceq \gamma \cdot I_N, \quad x \in R^N.$$

  Note that the maximal eigenvalue of symmetric matrix can reach its trace. Hence, in the worst case, the rate of convergence of the gradient method is the same as the rate of $RCDM$. However, the latter method has much more chances to accelerate.

3 Minimizing strongly convex functions

Let us estimate now the performance of $RCDM$ on strongly convex functions. Recall that f is called strongly convex on R^N with convexity parameter $\sigma = \sigma(f) > 0$ if for any x and y from R^N we have

$$f(y) \;\; \ge \;\; f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{1}{2} \sigma(f) \|y - x\|^2. \tag{3.1}$$

Minimizing both sides of this inequality in y, we obtain a useful bound

$$f(x) - f^* \;\; \le \;\; \tfrac{1}{2\sigma(f)} (\|\nabla f(x)\|^*)^2. \tag{3.2}$$

Theorem 2 *Let function $f(x)$ be strongly convex with respect to the norm $\|\cdot\|_{1-\alpha}$ with convexity parameter $\sigma_{1-\alpha} = \sigma_{1-\alpha}(f) > 0$. Then, for the sequence $\{x_k\}$ generated by $RCDM(\alpha, x_0)$ we have*

$$\phi_k - \phi^* \;\leq\; \left(1 - \tfrac{\sigma_{1-\alpha}(f)}{S_\alpha(f)}\right)^k (f(x_0) - f^*). \tag{3.3}$$

Proof:
In view of inequality (2.13), we have

$$f(x_k) - E_{i_k}(f(x_{k+1})) \;\geq\; \tfrac{1}{2S_\alpha}(\|\nabla f(x_k)\|^*_{1-\alpha})^2 \overset{(3.2)}{\geq} \tfrac{\sigma_{1-\alpha}}{S_\alpha}(f(x_k) - f^*).$$

It remains to compute the expectation in ξ_{k-1}. □

At this moment, we are able to prove only that the *expected quality* of the output of RCDM is good. However, in practice we are not going to run this method many times on the same problem. What is the probability that our single run can give us also a good result? In order to answer this question, we need to apply RCDM to a regularized objective of the initial problem. For the sake of simplicity, let us endow the spaces R^{n_i} with some Euclidean norms:

$$\|h^{(i)}\|^2_{(i)} \;=\; \langle B_i h^{(i)}, h^{(i)}\rangle, \quad h^{(i)} \in R^{n_i}, \tag{3.4}$$

where $B_i \succ 0$, $i = 1, \ldots, n$. We present the complexity results for two ways of measuring distances in R^N.

1. Let us use the norm $\|\cdot\|_1$:

$$\|h\|_1^2 \;=\; \sum_{i=1}^n L_i \langle B_i h^{(i)}, h^{(i)}\rangle \overset{\text{def}}{=} \langle B_L h, h\rangle, \quad h \in R^N. \tag{3.5}$$

Since this norm is Euclidean, the regularized objective function

$$f_\mu(x) \;=\; f(x) + \tfrac{\mu}{2}\|x - x_0\|_1^2$$

is strongly convex with respect to $\|\cdot\|_1$ with convexity parameter μ. Moreover,

$$S_0(f_\mu) \;=\; n$$

for any value of $\mu > 0$. Hence, by Theorem 2, method $RCDM(0, x_0)$ can quickly approach the value $f^*_\mu = \min\limits_{x \in R^N} f_\mu(x)$. The following result will be useful.

Lemma 3 *Let random point x_k be generated by $RCDM(0, x_0)$ as applied to function f_μ. Then*

$$E_{\xi_k}(\|\nabla f_\mu(x_k)\|^*_1) \;\leq\; \left[2(n+\mu)\cdot(f(x_0) - f^*)\cdot\left(1 - \tfrac{\mu}{n}\right)^k\right]^{1/2}. \tag{3.6}$$

Proof:
In view of Lemma 2, for any $h = (h^{(1)}, \ldots, h^{(n)}) \in R^N$ we have

$$f_\mu(x+h) \;\leq\; f(x) + \langle \nabla f(x), h\rangle + \tfrac{n}{2}\|h\|_1^2 + \tfrac{\mu}{2}\|x + h - x_0\|_1^2$$

$$\overset{(3.5)}{=} \; f_\mu(x) + \langle \nabla f_\mu(x), h\rangle + \tfrac{n+\mu}{2}\|h\|_1^2.$$

Thus, function f_μ has Lipschits-continuous gradient (e.g. (2.1.6) in Theorem 2.1.5, [6]) with constant $L(f_\mu) \stackrel{\rm def}{=} n+\mu$. Therefore, (see, for example, (2.1.7) in Theorem 2.1.5, [6]),

$$\frac{1}{2L(f_\mu)} \left(\|\nabla f_\mu(x)\|_1^*\right)^2 \;\leq\; f_\mu(x) - f_\mu^*, \quad x \in R^N.$$

Hence,

$$\begin{aligned} E_{\xi_k}(\|\nabla f_\mu(x_k)\|_1^*) &\leq E_{\xi_k}\left(\left[2(n+\mu)\cdot(f_\mu(x_k)-f_\mu^*)\right]^{1/2}\right) \\ &\leq \left[2(n+\mu)\cdot\left(E_{\xi_k}\left(f_\mu(x_k)\right)-f_\mu^*\right)\right]^{1/2} \\ &\stackrel{(3.3)}{\leq} \left[2(n+\mu)\cdot\left(1-\tfrac{\mu}{n}\right)^k\left(f(x_0)-f_\mu^*\right)\right]^{1/2}. \end{aligned}$$

It remains to note that $f_\mu^* \geq f^*$. □

Now we can estimate the quality of the random point x_k generated by $RCDM(0,x_0)$, taken as an approximate solution to problem (2.1). Let us fix the desired accuracy of the solution $\epsilon > 0$ and the confidence level $\beta \in (0,1)$.

Theorem 3 *Let us define* $\mu = \frac{\epsilon}{4R_1^2(x_0)}$, *and choose*

$$k \;\geq\; 1 + \frac{8nR_1^2(x_0)}{\epsilon}\ln\frac{2nR_1^2(x_0)}{\epsilon(1-\beta)}. \tag{3.7}$$

If the random point x_k *is generated by* $RCDM(0,x_0)$ *as applied to function* f_μ, *then*

$$\mathbf{Prob}\left(f(x_k)-f^* \leq \epsilon\right) \;\geq\; \beta.$$

Proof:
Note that $f(x_k) \leq f_\mu(x_k) \leq f_\mu(x_0) = f(x_0)$. Therefore, there exists $x_* \in X^*$ such that $\|x_k - x_*\|_1 \leq R_1(x_0) \stackrel{\rm def}{=} R$. Hence, $\|x_k - x_0\|_1 \leq \|x_k - x_*\|_1 + \|x_* - x_0\|_1 \leq 2R$. Since

$$\nabla f_\mu(x) \;=\; \nabla f(x) + \mu \cdot B_L(x - x_0), \tag{3.8}$$

we conclude that

$$\begin{aligned} \mathbf{Prob}\left(f(x_k)-f^* \geq \epsilon\right) &\leq \mathbf{Prob}\left(\|\nabla f(x_k)\|_1^* \cdot R \geq \epsilon\right) \\ &\stackrel{(3.8)}{\leq} \mathbf{Prob}\left(\|\nabla f_\mu(x_k)\|_1^* \cdot R + 2\mu R^2 \geq \epsilon\right) \\ &= \mathbf{Prob}\left(\|\nabla f_\mu(x_k)\|_1^* \geq \tfrac{\epsilon}{2R}\right). \end{aligned}$$

Now, using Chebyshev inequality, we obtain

$$\begin{aligned} \mathbf{Prob}\left(\|\nabla f_\mu(x_k)\|_1^* \geq \tfrac{\epsilon}{2R}\right) &\leq \tfrac{2R}{\epsilon}\cdot E_{\xi_k}\left(\|\nabla f_\mu(x_k)\|_1^*\right) \\ &\stackrel{(3.6)}{\leq} \tfrac{2R}{\epsilon}\cdot\left[2(n+\mu)\cdot\left(f(x_0)-f^*\right)\cdot\left(1-\tfrac{\mu}{n}\right)^k\right]^{1/2}. \end{aligned}$$

Since the gradient of function f is Lipschitz continuous with constant n, we get

$$f(x_0) - f^* \;\leq\; \tfrac{n}{2}R^2.$$

Taking into account that $(n+\mu)(1-\frac{\mu}{n})^k \leq n(1-\frac{\mu}{n})^{k-1}$, we obtain the following bound:

$$\mathbf{Prob}\left(\|\nabla f_\mu(x_k)\|_1^* \geq \tfrac{\epsilon}{2R}\right) \;\leq\; \tfrac{2nR^2}{\epsilon}\cdot\left(1-\tfrac{\mu}{n}\right)^{\frac{k-1}{2}} \;\leq\; \tfrac{2nR^2}{\epsilon}\cdot e^{-\frac{\mu(k-1)}{2n}} \;\leq\; 1-\beta.$$

This proves the statement of the theorem. □

For the current choice of norm (3.5), we can guarantee only $L(f) \leq n$. Therefore, the standard gradient method as applied to the problem (2.1) has the worst-case complexity bound of

$$O\left(\tfrac{nR_1^2(x_0)}{\epsilon}\right)$$

full-dimensional gradient iterations. Up to a logarithmic factor, this bound coincides with (3.7).

2. Consider now the following norm:

$$\|h\|_0^2 \;=\; \sum_{i=1}^n \langle B_i h^{(i)}, h^{(i)}\rangle \;\stackrel{\rm def}{=}\; \langle Bh, h\rangle, \quad h \in R^N, \tag{3.9}$$

Again, the regularized objective function

$$f_\mu(x) \;=\; f(x) + \tfrac{\mu}{2}\|x - x_0\|_0^2$$

is strongly convex with respect to $\|\cdot\|_0$ with convexity parameter μ. However, now

$$S_1(f_\mu) \;=\; \sum_{i=1}^n [L_i(f) + \mu] \;=\; S_1(f) + n\mu.$$

Since $L(f_\mu) = S_1(f) + \mu$, using the same arguments as in the proof of Lemma 3, we get the following bound.

Lemma 4 *Let random point x_k be generated by $RCDM(1, x_0)$ as applied to function f_μ. Then*

$$E_{\xi_k}(\|\nabla f_\mu(x_k)\|_0^*) \;\leq\; \left[2(S_1(f)+\mu)\cdot(f(x_0)-f^*)\cdot\left(1-\tfrac{\mu}{S_1(f)+n\mu}\right)^k\right]^{1/2}. \tag{3.10}$$

Using this lemma, we can prove the following theorem.

Theorem 4 *Let us define $\mu = \frac{\epsilon}{4R_0^2(x_0)}$, and choose*

$$k \;\geq\; 2\left[n + \tfrac{4S_1(f)R_0^2(x_0)}{\epsilon}\right]\cdot\left(\ln\tfrac{1}{1-\beta} + \ln\left(\tfrac{1}{2} + \tfrac{2S_1(f)R_0^2(x_0)}{\epsilon}\right)\right). \tag{3.11}$$

If the random point x_k is generated by $RCDM(1, x_0)$ as applied to function f_μ, then

$$\mathbf{Prob}\,(f(x_k) - f^* \leq \epsilon) \;\geq\; \beta.$$

Proof:
As in the proof of Theorem 3, we can prove that $\|x_k - x_0\|_0 \leq 2R \equiv 2R_0(x_0)$. Since

$$\nabla f_\mu(x) \;=\; \nabla f(x) + \mu \cdot B(x - x_0), \tag{3.12}$$

we conclude that

$$\begin{aligned}
\mathbf{Prob}\,(f(x_k) - f^* \geq \epsilon) \quad &\leq \quad \mathbf{Prob}\,(\|\nabla f(x_k)\|_0^* \cdot R \geq \epsilon) \\
&\overset{(3.12)}{\leq} \quad \mathbf{Prob}\,(\|\nabla f_\mu(x_k)\|_0^* \cdot R + 2\mu R^2 \geq \epsilon) \\
&= \quad \mathbf{Prob}\,\left(\|\nabla f_\mu(x_k)\|_0^* \geq \tfrac{\epsilon}{2R}\right).
\end{aligned}$$

Now, using Chebyshev inequality, we obtain

$$\begin{aligned}
\mathbf{Prob}\,\left(\|\nabla f_\mu(x_k)\|_0^* \geq \tfrac{\epsilon}{2R}\right) \;&\leq\; \tfrac{2R}{\epsilon} \cdot E_{\xi_k}\left(\|\nabla f_\mu(x_k)\|_0^*\right) \\
&\overset{(3.10)}{\leq}\; \tfrac{2R}{\epsilon} \cdot \left[2(S_1(f) + \mu) \cdot (f(x_0) - f^*) \cdot \left(1 - \tfrac{\mu}{S_1(f) + n\mu}\right)^k\right]^{1/2}.
\end{aligned}$$

Since the gradient of function f is Lipschitz continuous with constant $S_1(f)$, we get

$$f(x_0) - f^* \;\leq\; \tfrac{1}{2} S_1(f) R^2.$$

Thus, we obtain the following bound:

$$\begin{aligned}
\mathbf{Prob}\,\left(\|\nabla f_\mu(x_k)\|_1^* \geq \tfrac{\epsilon}{2R}\right) \;&\leq\; \tfrac{2R^2}{\epsilon} \cdot (S_1(f) + \mu) \cdot \left(1 - \tfrac{\mu}{S_1(f) + n\mu}\right)^{\frac{k}{2}} \\
&\leq\; \tfrac{2R^2}{\epsilon} \cdot (S_1(f) + \mu) \cdot e^{-\frac{\mu k}{2(S_1(f) + n\mu)}} \;\leq\; 1 - \beta.
\end{aligned}$$

This proves the statement of the theorem. □

We conclude the section with some remarks.

- The dependence of complexity bounds (3.7), (3.11) in the confidence level β is very moderate. Hence, even very high confidence level is easily achievable.
- The standard gradient method (GM) has the complexity bound of $O\left(\frac{L(f)R_0^2(x_0)}{\epsilon}\right)$ full-dimensional gradient iterations. Note that in the worst case $L(f)$ can reach $S_1(f)$. Hence, for the class of objective functions treated in Theorem 4, the worst-case complexity bounds of $RCDM(1, x_0)$ and GM are essentially the same. Note that RCDM needs a certain number of full cycles. But this number grows proportionally to the logarithms of accuracy and of the confidence level. Note that the computational cost of a single iteration of RCDM is very often much smaller than that of GM.
- Consider very sparse problems with the cost of n coordinate iterations being proportional to a single full-dimensional gradient iterations. Then the complexity bound of $RCDM(1, x_0)$ in terms of the groups of iterations of size n becomes $O^*\left(1 + \frac{S_1}{n} \cdot \frac{R_0^2(x_0)}{\epsilon}\right)$. As compared with the gradient method, in this complexity bound the largest eigenvalue of the Hessian is replaced by an estimate for its *average* eigenvalue.

4 Constrained minimization

Consider now the constrained minimization problem

$$\min_{x \in Q} \quad f(x), \tag{4.1}$$

where $Q = \bigotimes_{i=1}^{n} Q_i$, and the sets $Q_i \subseteq R^{n_i}$, $i = 1, \dots, n$, are closed and convex. Let us endow the spaces R^{n_i} with some Euclidean norms (3.4), and assume that the objective function f of problem (4.1) is convex and satisfies our main smoothness assumption (2.2).

We can define now the constrained coordinate update as follows:

$$\begin{aligned} u^{(i)}(x) &= \arg \min_{u^{(i)} \in Q_i} \left[\langle f_i'(x), u^{(i)} - x^{(i)} \rangle + \tfrac{L_i}{2} \|u^{(i)} - x^{(i)}\|_{(i)}^2 \right], \\ V_i(x) &= x + U_i^T (u^{(i)}(x) - x^{(i)}), \quad i = 1, \dots, n. \end{aligned} \tag{4.2}$$

The optimality conditions for these optimization problems can be written in the following form:

$$\langle f_i'(x) + L_i B_i (u^{(i)}(x) - x^{(i)}), u^i - u^{(i)}(x) \rangle \;\geq\; 0 \quad \forall u^{(i)} \in Q_i,\ i = 1, \dots, n. \tag{4.3}$$

Using this inequality for $u^{(i)} = x^{(i)}$, we obtain

$$\begin{aligned} f(V_i(x)) &\overset{(2.3)}{\leq} f(x) + \langle f_i'(x), u^{(i)}(x) - x^{(i)} \rangle + \tfrac{L_i}{2} \|u^{(i)}(x) - x^{(i)}\|_{(i)}^2 \\ &\overset{(4.3)}{\leq} f(x) + \langle L_i B_i (u^{(i)}(x) - x^{(i)}), x^i - u^{(i)}(x) \rangle + \tfrac{L_i}{2} \|u^{(i)}(x) - x^{(i)}\|_{(i)}^2. \end{aligned}$$

Thus,

$$f(x) - f(V_i(x)) \;\geq\; \tfrac{L_i}{2} \|u^{(i)}(x) - x^{(i)}\|_{(i)}^2, \quad i = 1, \dots, n. \tag{4.4}$$

Let us apply to problem (4.1) the *uniform* coordinate decent method (UCDM).

Method $UCDM(x_0)$

For $k \geq 0$ iterate:

1) Choose randomly i_k by uniform distribution on $\{1 \dots n\}$.

2) Update $x_{k+1} = V_{i_k}(x_k)$.

(4.5)

Theorem 5 *For any $k \geq 0$ we have*

$$\phi_k - f^* \;\leq\; \tfrac{n}{n+k} \cdot \left[\tfrac{1}{2} R_1^2(x_0) + f(x_0) - f^* \right].$$

If f is strongly convex in $\|\cdot\|_1$ with constant σ, then

$$\phi_k - f^* \;\leq\; \left(1 - \tfrac{2\sigma}{n(1+\sigma)}\right)^k \cdot \left(\tfrac{1}{2} R_1^2(x_0) + f(x_0) - f^* \right). \tag{4.6}$$

Proof:
We will use notation of Theorem 1. Let UCDM generate an implementation x_k of corresponding random variable. Denote

$$r_k^2 \;=\; \|x_k - x_*\|_1^2 \;=\; \sum_{i=1}^n L_i \langle B_i \left(x_k^{(i)} - x_*^{(i)}\right), x_k^{(i)} - x_*^{(i)} \rangle.$$

Then

$$r_{k+1}^2 = r_k^2 + 2L_{i_k}\langle B_{i_k}\left(u^{(i_k)}(x_k) - x_k^{(i_k)}\right), x_k^{(i_k)} - x_*^{(i_k)}\rangle + L_{i_k}\|u^{(i_k)}(x_k) - x_k^{(i_k)}\|_{(i_k)}^2$$

$$= r_k^2 + 2L_{i_k}\langle B_{i_k}\left(u^{(i_k)}(x_k) - x_k^{(i_k)}\right), u^{(i_k)}(x_k) - x_*^{(i_k)}\rangle - L_{i_k}\|u^{(i_k)}(x_k) - x_k^{(i_k)}\|_{(i_k)}^2$$

$$\stackrel{(4.3)}{\leq} r_k^2 + 2\langle f'_{i_k}(x_k), x_*^{(i_k)} - u^{(i_k)}(x_k)\rangle - L_{i_k}\|u^{(i_k)}(x_k) - x_k^{(i_k)}\|_{(i_k)}^2$$

$$= r_k^2 + 2\langle f'_{i_k}(x_k), x_*^{(i_k)} - x_k^{(i_k)}\rangle$$

$$-2\left[\langle f'_{i_k}(x_k), u^{(i_k)}(x_k) - x_k^{(i_k)}\rangle + \tfrac{1}{2}L_{i_k}\|u^{(i_k)}(x_k) - x_k^{(i_k)}\|_{(i_k)}^2\right]$$

$$\stackrel{(2.3)}{\leq} r_k^2 + 2\langle f'_{i_k}(x_k), x_*^{(i_k)} - x_k^{(i_k)}\rangle + 2\left[f(x_k) - f(V_{i_k}(x_k))\right].$$

Taking the expectation in i_k, we obtain

$$E_{i_k}\left(\tfrac{1}{2}r_{k+1}^2 + f(x_{k+1}) - f^*\right) \;\leq\; \tfrac{1}{2}r_k^2 + f(x_k) - f^* + \tfrac{1}{n}\langle \nabla f(x_k), x_* - x_k\rangle. \qquad (4.7)$$

Thus, for any $k \geq 0$ we have

$$\tfrac{1}{2}r_0^2 + f(x_0) - f^* \;\geq \phi_{k+1} - f^* + \tfrac{1}{n}\left[\langle \nabla f(x_0), x_0 - x_*\rangle + \sum_{i=1}^k E_{\xi_{i-1}}(\langle \nabla f(x_i), x_i - x_*\rangle)\right]$$

$$\geq \phi_{k+1} - f^* + \tfrac{1}{n}\sum_{i=0}^k (\phi_i - f^*) \stackrel{(4.4)}{\geq} \left(1 + \tfrac{k+1}{n}\right)(\phi_{k+1} - f^*).$$

Finally, let f be strongly convex in $\|\cdot\|_1$ with convexity parameter σ.[2)] Then, (e.g. Section 2.1 in [6]),

$$\langle \nabla f(x), x - x_*\rangle \geq f(x) - f^* + \tfrac{\sigma}{2}\|x - x_*\|_1^2 \geq \sigma\|x - x_*\|_1^2.$$

Define $\beta = \frac{2\sigma}{1+\sigma} \in [0, 1]$. Then, in view of inequality (4.7), we have

$$E_{i_k}\left(\tfrac{1}{2}r_{k+1}^2 + f(x_{k+1}) - f^*\right) \;\leq\; \tfrac{1}{2}r_k^2 + f(x_k) - f^*$$

$$-\tfrac{1}{n}\left[\beta(f(x_k) - f^* + \tfrac{\sigma}{2}r_k^2) + (1-\beta)\sigma r_k^2\right]$$

$$= \left(1 - \tfrac{2\sigma}{n(1+\sigma)}\right)\cdot\left(\tfrac{1}{2}r_k^2 + f(x_k) - f^*\right).$$

It remains to take expectation in ξ_{k-1}. □

[2)] In view of assumption (2.2), we always have $\sigma \leq 1$.

5 Accelerated coordinate descent

It is well known that the usual gradient method can be transformed in a faster scheme by applying an appropriate multistep strategy [5]. Let us show that this can be done also for random coordinate descent methods. Consider the following accelerated scheme as applied to the unconstrained minimization problem (2.1) with strongly convex objective function. We assume that the convexity parameter $\sigma = \sigma_1(f) \geq 0$ is known.

Method $ACDM(x_0)$

1. Define $v_0 = x_0$, $a_0 = \frac{1}{n}$, $b_0 = 2$.

2. For $k \geq 0$ iterate:

1) Compute $\gamma_k \geq \frac{1}{n}$ from equation $\gamma_k^2 - \frac{\gamma_k}{n} = (1 - \frac{\gamma_k \sigma}{n}) \frac{a_k^2}{b_k^2}$.
Set $\alpha_k = \frac{n - \gamma_k \sigma}{\gamma_k (n^2 - \sigma)}$, and $\beta_k = 1 - \frac{1}{n}\gamma_k \sigma$.

2) Select $y_k = \alpha_k v_k + (1 - \alpha_k) x_k$.

3) Choose $i_k = \mathcal{R}_0$ and update

$$x_{k+1} = T_{i_k}(y_k), \quad v_{k+1} = \beta_k v_k + (1 - \beta_k) y_k - \frac{\gamma_k}{L_{i_k}} U_{i_k} f_i'(y_k)^{\#}.$$

4) Set $b_{k+1} = \frac{b_k}{\sqrt{\beta_k}}$, and $a_{k+1} = \gamma_k b_{k+1}$.

(5.1)

For $n = 1$ and $\sigma = 0$ this method coincides with method [5]. Its parameters satisfy the following identity:

$$\gamma_k^2 - \frac{\gamma_k}{n} \;=\; \beta_k \frac{a_k^2}{b_k^2} \;=\; \frac{\beta_k \gamma_k}{n} \cdot \frac{1 - \alpha_k}{\alpha_k}. \tag{5.2}$$

Theorem 6 *For any $k \geq 0$ we have*

$$\begin{aligned} \phi_k - f^* &\leq \sigma \left[2\|x_0 - x^*\|_1^2 + \tfrac{1}{n^2}(f(x_0) - f^*) \right] \cdot \left[\left(1 + \tfrac{\sqrt{\sigma}}{2n}\right)^{k+1} - \left(1 - \tfrac{\sqrt{\sigma}}{2n}\right)^{k+1} \right]^{-2} \\ &\leq \left(\tfrac{n}{k+1}\right)^2 \cdot \left[2\|x_0 - x^*\|_1^2 + \tfrac{1}{n^2}(f(x_0) - f^*) \right]. \end{aligned} \tag{5.3}$$

Proof:
Let x_k and v_k be the implementations of corresponding random variables generated by $ACDM(x_0)$ after k iterations. Denote $r_k^2 = \|v_k - x_*\|_1^2$. Then using representation

$$v_k \;=\; y_k + \tfrac{1 - \alpha_k}{\alpha_k}(y_k - x_k),$$

we obtain

$$\begin{aligned} r_{k+1}^2 &= \|\beta_k v_k + (1-\beta_k)y_k - x^*\|_1^2 + \tfrac{\gamma_k^2}{L_{i_k}}\left(\|f'_{i_k}(y_k)\|^*_{(i_k)}\right)^2 \\ &\quad +2\gamma_k\langle f'_{i_k}(y_k), (x^* - \beta_k v_k - (1-\beta_k)y_k)^{(i_k)}\rangle \\ &\leq \|\beta_k v_k + (1-\beta_k)y_k - x^*\|_1^2 + 2\gamma_k^2(f(y_k) - f(T_{i_k}(y_k))) \\ &\quad +2\gamma_k\langle f'_{i_k}(y_k), \left(x^* - y_k + \tfrac{\beta_k(1-\alpha_k)}{\alpha_k}(x_k - y_k)\right)^{(i_k)}\rangle. \end{aligned}$$

Taking the expectation of both sides in i_k, and we obtain:

$$\begin{aligned} E_{i_k}(r_{k+1}^2) &\leq \beta_k r_k^2 + (1-\beta_k)\|y_k - x^*\|_1^2 + 2\gamma_k^2[f(y_k) - E_{i_k}(f(x_{k+1}))] \\ &\quad +2\tfrac{\gamma_k}{n}\langle \nabla f(y_k), x^* - y_k + \tfrac{\beta_k(1-\alpha_k)}{\alpha_k}(x_k - y_k)\rangle \\ &\leq \beta_k r_k^2 + (1-\beta_k)\|y_k - x^*\|_1^2 + 2\gamma_k^2[f(y_k) - E_{i_k}(f(x_{k+1}))] \\ &\quad +2\tfrac{\gamma_k}{n}[f^* - f(y_k) - \tfrac{1}{2}\sigma\|y_k - x^*\|_1^2 + \tfrac{\beta_k(1-\alpha_k)}{\alpha_k}(f(x_k) - f(y_k))] \\ &\overset{(5.2)}{=} \beta_k r_k^2 - 2\gamma_k^2 E_{i_k}(f(x_{k+1})) + 2\tfrac{\gamma_k}{n}[f^* + \tfrac{\beta_k(1-\alpha_k)}{\alpha_k}f(x_k)]. \end{aligned}$$

Note that

$$b_{k+1}^2 = \tfrac{1}{\beta_k}b_k^2, \quad a_{k+1}^2 = \gamma_k^2 b_{k+1}^2, \quad \gamma_k\tfrac{\beta_k(1-\alpha_k)}{n\alpha_k} = \tfrac{a_k^2}{b_{k+1}^2}.$$

Therefore, multiplying the last inequality by b_{k+1}^2, we obtain

$$b_{k+1}^2 E_{i_k}(r_{k+1}^2) \leq b_k^2 r_k^2 - 2a_{k+1}^2(E_{i_k}(f(x_{k+1})) - f^*) + 2a_k^2(f(x_k) - f^*).$$

Taking now the expectation of both sides of this inequality in ξ_{k-1}, we get

$$\begin{aligned} 2a_{k+1}^2(\phi_{k+1} - f^*) + b_{k+1}^2 E_{\xi_k}(r_{k+1}^2) &\leq 2a_k^2(\phi_k - f^*) + b_k^2 r_k^2 \\ &\leq 2a_0^2(f(x_0) - f^*) + b_0^2\|x_0 - x^*\|_1^2. \end{aligned}$$

It remains to estimate the growth of coefficients a_k and b_k. We have:

$$b_k^2 = \beta_k b_{k+1}^2 = (1 - \tfrac{\sigma}{n}\gamma_k)\, b_{k+1}^2 = \left(1 - \tfrac{\sigma}{n}\tfrac{a_{k+1}}{b_{k+1}}\right) b_{k+1}^2.$$

Thus, $\frac{\sigma}{n}a_{k+1}b_{k+1} \leq b_{k+1}^2 - b_k^2 \leq 2b_{k+1}(b_{k+1} - b_k)$, and we conclude that

$$b_{k+1} \geq b_k + \tfrac{\sigma}{2n}a_k. \tag{5.4}$$

On the other hand, $\frac{a_{k+1}^2}{b_{k+1}^2} - \frac{a_{k+1}}{nb_{k+1}} \overset{(5.2)}{=} \frac{\beta_k a_k^2}{b_k^2} = \frac{a_k^2}{b_{k+1}^2}$. Therefore,

$$\tfrac{1}{n}a_{k+1}b_{k+1} \leq a_{k+1}^2 - a_k^2 \leq 2a_{k+1}(a_{k+1} - a_k),$$

and we obtain

$$a_{k+1} \;\geq\; a_k + \tfrac{1}{2n} b_k. \tag{5.5}$$

Further, denoting $Q_1 = 1 + \frac{\sqrt{\sigma}}{2n}$ and $Q_2 = 1 - \frac{\sqrt{\sigma}}{2n}$ and using inequalities (5.4) and (5.5), it is easy to prove by induction that

$$a_k \;\geq\; \tfrac{1}{\sqrt{\sigma}} \left[Q_1^{k+1} - Q_2^{k+1} \right], \quad b_k \;\geq\; Q_1^{k+1} + Q_2^{k+1}.$$

Finally, using trivial inequality $(1+t)^k - (1-t)^k \geq 2kt$, $t \geq 0$, we obtain

$$Q_1^{k+1} - Q_2^{k+1} \geq \tfrac{k+1}{n} \sqrt{\sigma}.$$

□

The rate of convergence (5.3) of ACDM is much better than the rate (2.14). However, for some applications (e.g. Section 6.3), the complexity of one iteration of the accelerated scheme is rather high since for computing y_k it needs to operate with full-dimensional vectors.

6 Implementation details and numerical test

6.1 Dynamic adjustment of Lipschitz constants

In RCDM (2.6) we use the valid upper bounds for Lipschitz constants $\{L_i\}_{i=1}^n$ of the directional derivatives. For some applications (e.g. Section 6.3) this information is easily available. However, for more complicated functions we need to apply a dynamic adjustment procedure for finding appropriate bounds. Let us estimate the efficiency of a simple backtracking strategy with restore (e.g. [7]) inserted in $RCDM(0, x_0)$. As we have already discussed, such a strategy *should not* be based on computation of the function values.

Consider the Random Adaptive Coordinate Decent Method. For the sake of notation,

we assume that $N = n$.

Method $RACDM(x_0)$
Setup: lower bounds $\hat{L}_i := L_i^0 \in (0, L_i]$, $i = 1, \dots, n$.
For $k \geq 0$ **iterate:** **1)** Choose $i_k = \mathcal{R}_0$. **2)** Set $x_{k+1} := x_k - \hat{L}_{i_k}^{-1} \cdot \nabla_{i_k} f(x_k) \cdot e_{i_k}$. **while** $\nabla_{i_k} f(x_k) \cdot \nabla_{i_k} f(x_{k+1}) < 0$ **do** $\left\{ \hat{L}_{i_k} := 2\hat{L}_{i_k}, \quad x_{k+1} := x_k - \hat{L}_{i_k}^{-1} \cdot \nabla_{i_k} f(x_k) \cdot e_{i_k} \right\}$ **3)** Set $L_{i_k} := \frac{1}{2} L_{i_k}$.

(6.1)

Theorem 7 *1. At the beginning of each iteration, we have*

$$\hat{L}_i \leq L_i, \quad i = 1, \dots, n.$$

2. RACDM has the following rate of convergence:

$$\phi_k - f^* \leq \frac{8nR_1^2(x_0)}{16+3k}, \quad k \geq 0. \tag{6.2}$$

3. After iteration k, the total number N_k of computations of directional derivatives in method (6.1) satisfies inequality

$$N_k \leq 2(k+1) + \sum_{i=1}^{n} \log_2 \frac{L_i}{L_i^0}. \tag{6.3}$$

Proof:
For proving the first statement, we assume that at the entrance to the internal cycle in method (6.1), we have $\hat{L}_{i_k} \leq L_{i_k}$. If during this cycle we get $\hat{L}_{i_k} > L_{i_k}$ then immediately

$$|\nabla_{i_k} f(x_{k+1}) - \nabla_{i_k} f(x_k)| \overset{(2.2)}{\leq} \hat{L}_{i_k}^{-1} L_{i_k} |\nabla_{i_k} f(x_k)| < |\nabla_{i_k} f(x_k)|.$$

In this case, the termination criterion is satisfied. Thus, during the internal cycle

$$\hat{L}_{i_k} \leq 2L_{i_k}. \tag{6.4}$$

Therefore, after execution of Step 3, we have again $\hat{L}_{i_k} \leq L_{i_k}$.

Further, the internal cycle is terminated with $\hat{L}_{i_k}$ satisfying inequality (6.4). Therefore, using inequality (2.1.17) in [6], we have:

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \nabla_{i_k} f(x_{k+1}) \cdot \left(x_k^{(i_k)} - x_{k+1}^{(i_k)} \right) + \tfrac{1}{2\hat{L}_{i_k}} \left(\nabla_{i_k} f(x_{k+1}) - \nabla_{i_k} f(x_k) \right)^2 \\ &= \hat{L}_{i_k}^{-1} \cdot \nabla_{i_k} f(x_{k+1}) \cdot \nabla_{i_k} f(x_k) + \tfrac{1}{2\hat{L}_{i_k}} \left(\nabla_{i_k} f(x_{k+1}) - \nabla_{i_k} f(x_k) \right)^2 \\ &\overset{(6.4)}{\geq} \tfrac{1}{2\hat{L}_{i_k}} \left[\left(\nabla_{i_k} f(x_{k+1}) \right)^2 + \left(\nabla_{i_k} f(x_k) \right)^2 - \nabla_{i_k} f(x_{k+1}) \cdot \nabla_{i_k} f(x_k) \right] \\ &\geq \tfrac{3}{8\hat{L}_{i_k}} \left(\nabla_{i_k} f(x_k) \right)^2 . \end{aligned}$$

Thus, we obtain the following bound:

$$f(x_k) - E_{i_k}(f(x_{k+1})) \geq \tfrac{3}{8n} \left(\|\nabla f(x_k)\|_1^* \right)^2 ,$$

(compare with (2.13)). Now, using the same arguments as in the proof of Theorem 1, we can prove that

$$\phi_k - \phi_{k+1} \geq \tfrac{1}{C} (\phi_k - f^*)^2$$

with $C = \frac{8n}{3} R_1^2(x_0)$. And we conclude that

$$\tfrac{1}{\phi_k - f^*} \geq \tfrac{1}{f(x_0) - f^*} + \tfrac{k}{C} \geq \tfrac{2}{n R_1^2(x_0)} + \tfrac{3k}{8n R_1^2(x_0)}.$$

Finally, let us find an upper estimate for N_k. Denote by K_i a subset of iteration numbers $\{0, \ldots, k\}$, at which the index i was active. Denote by $p_i(j)$ the number of computations of ith partial derivative at iteration $j \in K_i$. If $\hat{L}_i$ is the current estimate of the Lipschitz constant L_i in the beginning of iteration j, and $\hat{L}_i'$ is the value of this estimate in the end of this iteration, then these values are related as follows:

$$\hat{L}_i' = \tfrac{1}{2} \cdot 2^{p_i(j)-1} \cdot \hat{L}_i.$$

In other words, $p_i(j) = 2 + \log_2 \left[\hat{L}_i' / \hat{L}_i \right]$. Taking into account the statement of Item 1, we obtain the following estimate for M_i, the total number of computations of ith partial in the first k iterations:

$$M_i \leq 2 \cdot |K_i| + \log_2 \tfrac{L_i}{L_i^0}.$$

It remains to note that $\sum\limits_{i=1}^{n} |K_i| = k + 1$. □

Note that the similar technique can be used also in the accelerated version (6.1). For that, we need to choose its parameters from the equations

$$\begin{aligned} \tfrac{4}{3}\gamma_k^2 - \tfrac{\gamma_k}{n} &= \left(1 - \tfrac{\gamma_k \sigma}{n}\right) \tfrac{a_k^2}{b_k^2}, \\ \beta_k &= 1 - \tfrac{\gamma_k \sigma}{n}, \\ \tfrac{1-\alpha_k}{\alpha_k} &= \tfrac{n}{\beta_k} \cdot \left(\tfrac{4}{3}\gamma_k - \tfrac{1}{n} \right). \end{aligned} \tag{6.5}$$

This results in a minor change in the rate of convergence (5.3).

6.2 Random counters

For problems of very big size, we should treat carefully any operation with multidimensional vectors. In RCD-methods, there is such an operation, which must be fulfilled at each step of the algorithms. This is the random generation of active coordinates. In a straightforward way, this operation can be implemented in $O(n)$ operations. However, for huge-scale problems this complexity can be prohibitive. Therefore, before presenting our computational results, we describe a strategy for generating random coordinates with complexity $O(\ln n)$ operations.

Given the values L_i^α, $i = 1, \dots, n$, we need to generate efficiently random integer numbers $i \in \{1, \dots, n\}$ with probabilities

$$\mathbf{Prob}\,[i = k] \;=\; L_k^\alpha \Big/ \left[\sum_{j=1}^{n} L_j^\alpha\right], \quad k = 1, \dots, n.$$

Without loss of generality, we assume that $n = 2^m$. Define $m+1$ vectors $P_k \in R^{2^{m-k}}$, $k = 0, \dots, m$, as follows:

$$\begin{aligned} P_0^{(i)} &= L_i^\alpha,\ i = 1, \dots, n. \\ P_k^{(i)} &= P_{k-1}^{(2i)} + P_{k-1}^{(2i-1)},\ i = 1, \dots, 2^{m-k}, \quad k = 1, \dots, m. \end{aligned} \tag{6.6}$$

Clearly, $P_m^{(1)} = S_\alpha$. Note that the preliminary computations (6.6) need $\frac{n}{2}$ additions.

Let us describe now our random generator.

1. Choose $i = 1$ in P_m.

2. **For** $k = m$ **down to** 1 **do:**

Let the element i of P_k be chosen. Choose in P_{k-1}

either $2i$ or $2i-1$ with probabilities $\frac{P_{k-1}^{(2i)}}{P_k^{(i)}}$ or $\frac{P_{k-1}^{(2i-1)}}{P_k^{(i)}}$. (6.7)

Clearly, this procedure implements correctly the random counter $\mathcal{R}_\alpha$. Note that its complexity is $O(\ln n)$ operations. In the above form, its execution requires generating m random numbers. However, a simple modification can reduce this amount up to one. Note also, that corrections of vectors P_k, $k = 0, \dots, m$ due to the change of a single entry in the initial data needs $O(\ln n)$ operations.

6.3 Numerical test

Let us describe now our test problem (sometimes it is called the *Google problem*). Let $E \in R^{n \times n}$ be an incidence matrix of a graph. Denote $e = (1, \dots, 1)^T$ and

$$\bar{E} = E \cdot \mathrm{diag}\,(E^T e)^{-1}.$$

Since, $\bar{E}^T e = e$, it is a stochastic matrix. Our problem consists in finding a right maximal eigenvector of the matrix $\bar{E}$:

$$\text{Find } x^* \geq 0: \quad \bar{E}x^* = x^*, \ \langle e, x \rangle = 1.$$

Clearly, this problem can be rewritten in an optimization form:

$$f(x) \stackrel{\text{def}}{=} \tfrac{1}{2}\|\bar{E}x - x\|^2 + \tfrac{\gamma}{2}[\langle e, x \rangle - 1]^2 \quad \rightarrow \quad \min_{x \in R^n}, \tag{6.8}$$

where $\gamma > 0$ is a penalty parameter for the equality constraint, and the norm $\|\cdot\|$ is Euclidean. If the degree of each node in the graph is small, then the computation of partial derivatives of function f is cheap. Hence, we can apply to (6.8) RCDM even if the size of the matrix $\bar{E}$ is very big.

In our numerical experiments we applied $RCDM(1,0)$ to a randomly generated graph with average node degree p. The termination criterion was

$$\|\bar{E}x - x\| \leq \epsilon \cdot \|x\|$$

with $\epsilon = 0.01$. In the table below, k denotes the total number of groups by n coordinate iterations. The computations were performed on a standard Pentium-4 computer with frequency 1.6GHz.

n	p	γ	k	Time(s)
65536	10	$\frac{1}{n}$	47	7.41
	20		30	5.97
	10	$\frac{1}{\sqrt{n}}$	65	10.5
	20		39	7.84
262144	10	$\frac{1}{n}$	47	42.7
	20		32	39.1
	10	$\frac{1}{\sqrt{n}}$	72	76.5
	20		45	62.0
1048576	10	$\frac{1}{n}$	49	247
	20		31	240
	10	$\frac{1}{\sqrt{n}}$	82	486
	20		64	493

We can see that the number of n-iteration groups grows very moderately with the dimension of the problem. The increase of factor γ also does not create for RCDM significant difficulties. Note that in the standard Euclidean norm we have $L(f) \approx \gamma n$. Thus, for the black-box gradient methods the problems with $\gamma = \frac{1}{\sqrt{n}}$ are very difficult. Note also that the accelerated scheme (6.1) is not efficient on problems of this size. Indeed, each coordinate iteration of this method needs an update of n-dimensional vector. Therefore, one group of n iterations takes at least $n^2 \approx 10^{12}$ operations (this is for the maximal dimension in the above table). For our computer, this amount of computations takes at least 20 minutes.

Note that the dominance of RCDM on sparse problems can be supported by comparison of the efficiency estimates. Let us put in one table the complexity results related to RCDM (2.6), accelerated scheme ACDM (5.1), and the fast gradient method (FGM)

working with the full gradient (e.g. Section 2.2 in [6]). Denote by T the complexity of computation of single directional derivative, and by F complexity of computation of the function value. We assume that $F = nT$. Note that this can be true even for dense problems (e.g. quadratic functions). We will measure distances in $\|\cdot\|_1$. For this metric, denote by γ the Lipschitz constant for the gradient of the objective function. Note that

$$1 \leq \gamma \leq n.$$

In the table below, we compare the cost of iteration, cost of the oracle and the iteration complexity for these three methods.

	RCDM	*ACDM*	*FGM*
Iteration	1	n	n
Oracle	T	T	F
Complexity	$n\frac{R^2}{\epsilon}$	$n\frac{R}{\sqrt{\epsilon}}$	$\frac{\sqrt{\gamma}R}{\sqrt{\epsilon}}$
Total	$(F+n)\frac{R^2}{\epsilon}$	$(F+n^2)\frac{R}{\sqrt{\epsilon}}$	$(F+n)\frac{\sqrt{\gamma}R}{\sqrt{\epsilon}}$

(6.9)

We can see that RCDM is better than FGM if $\frac{R}{\sqrt{\epsilon}} < \sqrt{\gamma}$. On the other hand, FGM is better than ACDM if $F < \frac{n^2}{\sqrt{\gamma}}$. For our test problem, both conditions are satisfied.

References

[1] A. Auslender. *Optimisation Méthodes Numériques*. Masson, Paris (1976).

[2] D. Bertsekas. *Nonlinear Programming*. 2nd edition, Athena Scientific, Belmont (1999).

[3] Z.Q. Luo, P. Tseng. On the convergence rate of dual ascent methods for linearly constrained convex minimization. *Mathematics of Operations Research*, **18**(2), 846-867 (1993).

[4] A. Nemirovsky, D. Yudin. *Problem complexity and method efficiency in optimization*. John Willey & Sons, Somerset, NJ (1983).

[5] Yu. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(\frac{1}{k^2})$. *Doklady AN SSSR* (translated as Soviet Math. Docl.), **269**(3), 543-547 (1983).

[6] Yu. Nesterov. *Introductory Lectures on Convex Optimization*. Kluwer, Boston, 2004.

[7] Yu. Nesterov. Gradient methods for minimizing composite objective function. CORE Discussion Paper 2007/76, (2007).

[8] P. Tseng. Convergence of a block coordinate descent methods for nondifferentiable minimization. JOTA, **109**(3), 475-494 (2001).