J Optim Theory Appl (2014) 162:892–919
DOI 10.1007/s10957-013-0491-5

# The 2-Coordinate Descent Method for Solving Double-Sided Simplex Constrained Minimization Problems

**Amir Beck**

Received: 11 November 2012 / Accepted: 19 November 2013 / Published online: 3 December 2013


**Abstract** This paper considers the problem of minimizing a continuously differentiable function with a Lipschitz continuous gradient subject to a single linear equality constraint and additional bound constraints on the decision variables. We introduce and analyze several variants of a *2-coordinate descent* method: a block descent method that performs an optimization step with respect to only two variables at each iteration. Based on two new optimality measures, we establish convergence to stationarity points for general nonconvex objective functions. In the convex case, when all the variables are lower bounded but not upper bounded, we show that the sequence of function values converges at a sublinear rate. Several illustrative numerical examples demonstrate the effectiveness of the method.

**Keywords** Nonconvex optimization · Simplex-type constraints · Block descent method · Rate of convergence

## 1 Introduction

Block descent algorithms are methods in which an optimization problem is solved by performing at each iteration a minimization step with respect to a small number of decision variables while keeping all other variables fixed. This kind of approach is also referred to in the literature as a "decomposition" approach. One of the first variable decomposition methods for solving general minimization problems was the so-called alternating minimization method [1, 2], which is based on successive global minimization with respect to each component vector in a cyclic order. This fundamental

Communicated by Gianni Di Pillo.

A. Beck (✉)
Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 32000, Israel
e-mail: becka@ie.technion.ac.il

method appears in the literature under various names such as the block-nonlinear Gauss-Seidel method or the block coordinate descent method (see, e.g., [3–11]). The convergence of the method was extensively studied in the aforementioned papers under various assumptions such as the existence of error bounds, strict/strong convexity with respect to each block, or uniqueness of minimizers with respect to each block.

Other block descent methods, that do not require a full minimization step at each iteration, are for example those who employ at each iteration a gradient projection step with respect to the chosen indices subset. These methods have a clear advantage over alternating minimization when exact minimization with respect to each of the component blocks is not an easy task. In [12] Luo and Tseng studied the convergence of such a block gradient descent method for specially structured convex problems with strict convexity assumptions on some of the elements of the objective function and with box constraints. More general descent directions were studied by Polak et al. in [13] and also by Sargent and Sebastian in [14]. Convergence to stationary points was established by Tseng and Yun [15] for the nonconvex case when the objective function has separable nonsmooth components. There seem to be only a few results in the literature on the rate of convergence of the sequence of function values in the absence of assumptions such as strong/strict convexity or the existence of an error bound. One such result was obtained by Nesterov in [16], where he showed that if the blocks are randomly chosen, then a sublinear rate of convergence of the expected sequence of function values can be derived. Later on in [17] it was shown that if the block selection is done in a cyclic manner, a sublinear rate of convergence can be derived.

All the mentioned works assume that the problem is either unconstrained or consists of block-wise constraints. When additional linear constraints are imposed, the block-wise structure collapses and a different line of analysis is required. The first algorithms for this class of problems were proposed for solving the dual of the support vector machine (SVM) optimization problem [18]. Hence, they are defined for the subclass of quadratic convex functions. In the quadratic case, part of the strength of 2-coordinate descent methods is in the fact that exact minimization can be analytically performed. Platt's sequential minimal optimization (SMO) algorithm [19] was the first 2-coordinate descent method with exact minimization. The simplicity and practical efficiency of the method motivated a vast amount of theoretical and practical research on the convergence of the method, as well as modifications and improvements. Later, Keerthi et al. in [20, 21] proposed a modified SMO method based on a kind of "most descent" choice of the coordinates. The so-called $\text{SVM}^{\text{light}}$ method proposed in [22] uses the same index selection strategy; in fact, it is more general in the sense that the method performs the optimization with respect to $q$ chosen variables—$q$ being an even positive integer. Convergence was proved in [23, 24] for the quadratic convex problem. A decomposition method based on the same selection rule for a block of variables was defined for more general smooth functions in [25], also allowing inexact minimization of the subproblems and odd values of $q$. Later methods for the more general (and even nonconvex) smooth problem were proposed; these methods are based on different selection rules and use either first order information [25–27], and might not require any ordering (e.g., cyclic selection rule) [28], or second order rules [29, 30]. All these methods are proved to have asymptotic convergence under quite mild assumptions. In [31–33], methods based on the projected

gradient (PG) direction have been proposed. Later, Tseng and Yun [34] studied the convergence of a coordinate (projected) gradient descent method for solving the general model of linearly constrained smooth minimization problems, which includes the dual SVM problem as a special case. A possible distributed version of a block-type method has also been proposed in [35].

In this paper, we consider the problem of minimizing a continuously differentiable function subject to a single linear constraint with additional bound constraints on the decision variables; a precise definition of the problem is presented in Sect. 2. We consider several variations of a block descent method, which we call *the 2-coordinate descent method*, that involves at each iteration the (possible approximated) solution of the optimization problem with respect to only two variables while keeping all other variables fixed; the two-dimensional minimization subproblems can also be reduced into one-dimensional optimization problems. After discussing several necessary mathematical preliminaries in Sect. 3, that lay the ground for the basic terminology in the paper, we present and analyze in Sect. 4 two "optimality measures" which will be the basis for the construction and analysis of the 2-coordinate descent methods devised in Sect. 5. The different variants of the method are dictated by the index selection strategy (i.e., which indices are chosen at iteration) and by the step-size selection strategy (full or partial). We show the convergence of the corresponding optimality measures in the nonconvex case. In the convex case, when all the variables are lower bounded but not upper bounded (as in the case of the unit simplex), we show in Sect. 6 a sublinear rate of convergence of the sequence of function values. The paper ends in Sect. 7, where several numerical experiments demonstrate the potential of the method.

## 2 Problem Formulation and Setting

We begin by describing some of the notation that will be used throughout the paper. For a positive integer $i$, the vector $\mathbf{e}_i$ denotes the $i$-th canonical basis vector, that is, it consists of zeros except for the $i$-th entry which is equal to one. For a differentiable function $f$, the gradient is denoted by $\nabla f$ and we use the notation $\nabla_i f$ for the $i$-th partial derivative. For a closed and convex set $X$, the orthogonal projection onto the set $X$ is denoted by $P_X(\cdot)$ and defined by $P_X(\mathbf{y}) := \operatorname{argmin}\{\|\mathbf{x}-\mathbf{y}\| : \mathbf{x} \in X\}$. Vectors are denoted by boldface lowercase letters, and matrices by boldface uppercase letters.

The main problem we consider in this paper is

$$\text{(P)} \quad \min\left\{ f(\mathbf{x}) : \mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}} \right\},$$

where the feasible set is given by:

$$\Delta_n^{K,\mathbf{l},\mathbf{u}} := \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{i=1}^n x_i = K, l_i \le x_i \le u_i \right\}.$$

Here $K \in \mathbb{R}$ and for each $i$, $u_i \in \mathbb{R} \cup \{\infty\}$ and $l_i \in \mathbb{R} \cup \{-\infty\}$. We will call such a set *a double-sided simplex* set. We assume that $l_i < u_i$ for all $i$ and, in order to ensure

feasibility, we also assume that $\sum_{i=1}^n l_i \le K \le \sum_{i=1}^n u_i$. Note that we do allow some or all of the upper and lower bounds to be infinite, and we will use the usual arithmetic of infinite numbers (e.g., $\infty + a = \infty$ for all $a \in \mathbb{R}$).

We will also be interested in the special case when there are no finite upper bounds, that is, $u_i = \infty$ for all $i = 1, 2, \ldots, n$. In this case, the set will be called *a one-sided simplex* set and is given by (the superscript $\mathbf{u}$ can be omitted in this case):

$$\Delta_n^{K,\mathbf{l}} := \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{i=1}^n x_i = K, x_i \ge l_i, i = 1, \ldots, n \right\}.$$

When $K = 1$ and $\mathbf{l} = \mathbf{0}$, the one-sided simplex $\Delta_n^{1,\mathbf{0}}$ is called the *unit simplex*:

$$\Delta_n := \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{i=1}^n x_i = 1, x_1, \ldots, x_n \ge 0 \right\}.$$

Despite the fact that a one-sided simplex set is also a two-sided simplex set, we will see that in some parts of the analysis it is worthwhile to take special care for this subclass of problems, since the results for the one-sided setting are sometimes simpler and stronger than those that can be obtained in the more general two-sided setting.

The following set of assumptions is made throughout the paper.

**Assumption 1**

- *The objective function* $f : \Delta_n^{K,\mathbf{l},\mathbf{u}} \to \mathbb{R}$ *is continuously differentiable with Lipschitz continuous gradient with constant* $L$ *over* $\Delta_n^{K,\mathbf{l},\mathbf{u}}$, *that is*,

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \le L\|\mathbf{x} - \mathbf{y}\| \quad \textit{for all } \mathbf{x}, \mathbf{y} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}.$$

- *Problem* (P) *is solvable*, *that is*, *it has a nonempty optimal solution set denoted by* $X^*$. *The optimal value will be denoted by* $f^*$.

There are numerous minimization problems over two-sided simplex domains. Among them is the dual problem associated with the problem of training an SVM, which is a convex quadratic programming problem [18]. Another problem of type (P) is the standard quadratic programming problem (StQP), which requires the minimization of a quadratic function over the unit simplex, which arises, for example, in the Markowitz portfolio optimization problem [36] and as the continuous formulation of combinatorial problems [37]. Another interesting problem is the dual of the Chebyshev center problem [38]. Some of these examples will be discussed in detail in Sect. 7.

### 2.1 General Linear Constraint

A seemingly more general problem than (P) is

$$\min \left\{ g(\mathbf{y}) : \sum_{i=1}^n a_i y_i = K, m_i \le y_i \le M_i, i = 1, 2, \ldots, n \right\}, \tag{1}$$

where $a_i \neq 0$ for all $i = 1, 2, \ldots, n$ and $m_i, M_i, i = 1, 2, \ldots, n$ are such that the feasible set is nonempty. However, problem (1) is not actually more general since we can easily recast it in the form of model (P) by making the change of variables $x_i = a_i y_i, i = 1, 2, \ldots, n$, resulting in the problem:

$$\min\left\{ g\left(\frac{x_1}{a_1}, \ldots, \frac{x_n}{a_n}\right) : \sum_{i=1}^{n} x_i = K, \tilde{m}_i \leq x_i \leq \tilde{M}_i, i = 1, 2, \ldots, n \right\}, \tag{2}$$

where

$$\tilde{m}_i = \begin{cases} \frac{m_i}{a_i} & a_i > 0 \\ \frac{M_i}{a_i} & a_i < 0, \end{cases} \qquad \tilde{M}_i = \begin{cases} \frac{M_i}{a_i} & a_i > 0 \\ \frac{m_i}{a_i} & a_i < 0. \end{cases}$$

Problem (2) fits model (P) with $l_i = \tilde{m}_i, u_i = \tilde{M}_i$ and $f(\mathbf{x}) := g(\frac{x_1}{a_1}, \ldots, \frac{x_n}{a_n})$. For simplicity of presentation we will analyze the model (P); however, as will be shown in the sequel, the results and algorithms can be described also for the more general model (1) by using the simple transformation just described.

## 3 Mathematical Preliminaries

Our main objective is to construct a block descent method that performs a minimization step with respect to two variables at each iteration. The coupling constraint (that is, the constant sum constraint) prevents the development of an algorithm that performs a minimization with respect to only *one* variable at each iteration. We will therefore be interested in the restriction of the objective function $f$ on feasible directions consisting of only two nonzero components.

Since we are interested in the feasible directions with two nonzero components, we will define for any $\mathbf{z} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$ and any two different indices the following function:

$$\phi_{i,j,\mathbf{z}}(t) := f\big(\mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j)\big), \quad t \in I_{i,j,\mathbf{z}},$$

where the interval $I_{i,j,\mathbf{z}}$ comprises the feasible steps, that is,

$$I_{i,j,\mathbf{z}} := \big\{t : \mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j) \in \Delta_n^{K,\mathbf{l},\mathbf{u}}\big\}.$$

A simple computation shows that $I_{i,j,\mathbf{z}}$ can be written explicitly as

$$I_{i,j,\mathbf{z}} = \big[\max\{l_i - z_i, z_j - u_j\}, \min\{u_i - z_i, z_j - l_j\}\big].$$

The derivative of $\phi_{i,j,\mathbf{z}}(t)$ is given by

$$\phi'_{i,j,\mathbf{z}}(t) = \nabla_i f\big(\mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j)\big) - \nabla_j f\big(\mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j)\big). \tag{3}$$

The Lipschitz continuity of $\nabla f$ implies the Lipschitz continuity of $\phi'_{i,j,\mathbf{z}}(t)$ over $I_{i,j,\mathbf{z}}$, and for any $i, j$ we will denote by $L_{i,j}$ the constant for which:

$$\big|\phi'_{i,j,\mathbf{z}}(t) - \phi'_{i,j,\mathbf{z}}(s)\big| \leq L_{i,j}|t - s|, \quad \text{for all } \mathbf{z} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}, t, s \in I_{i,j,\mathbf{z}}.$$

The constants $L_{i,j}$ will be called *the local Lipschitz constants,* and they satisfy the following bound:

$$L_{i,j} \leq 2L.$$

Indeed, note that by (3) we have that

$$\begin{aligned}\left|\phi'_{i,j,\mathbf{z}}(t) - \phi'_{i,j,\mathbf{z}}(s)\right| &= \left|(\mathbf{e}_i - \mathbf{e}_j)\nabla f(\mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j) - (\mathbf{e}_i - \mathbf{e}_j)\nabla f(\mathbf{z} + s(\mathbf{e}_i - \mathbf{e}_j)\right| \\ &\leq \|\mathbf{e}_i - \mathbf{e}_j\| \left\|\nabla f(\mathbf{z} + t(\mathbf{e}_i - \mathbf{e}_j) - \nabla f(\mathbf{z} + s(\mathbf{e}_i - \mathbf{e}_j)\right\| \\ &\leq \|\mathbf{e}_i - \mathbf{e}_j\| \cdot L|t - s| \cdot \|\mathbf{e}_i - \mathbf{e}_j\| = L\|\mathbf{e}_i - \mathbf{e}_j\|^2 \cdot |t - s| \\ &= 2L|t - s|.\end{aligned}$$

*Example 3.1* Suppose that the objective function is a quadratic function of the form

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{b}^T\mathbf{x},$$

where $\mathbf{b} \in \mathbb{R}^n$ and $\mathbf{Q} = \mathbf{Q}^T \in \mathbb{R}^{n\times n}$. Then denoting $\mathbf{d} = \mathbf{e}_i - \mathbf{e}_j$, after some rearrangement of terms, we have that

$$\begin{aligned}\phi_{i,j,\mathbf{z}}(t) &= f(\mathbf{z} + t\mathbf{d}) \\ &= \frac{1}{2}\big(\mathbf{d}^T\mathbf{Q}\mathbf{d}\big)t^2 + \mathbf{d}^T(\mathbf{Q}\mathbf{z} + \mathbf{b})t + \frac{1}{2}\mathbf{z}^T\mathbf{Q}\mathbf{z} + \mathbf{b}^T\mathbf{z}.\end{aligned}$$

Therefore,

$$\phi'_{i,j,\mathbf{z}}(t) = \big(\mathbf{d}^T\mathbf{Q}\mathbf{d}\big)t + \mathbf{d}^T(\mathbf{Q}\mathbf{z} + \mathbf{b}) = (Q_{ii} + Q_{jj} - 2Q_{ij})t + \mathbf{d}^T(\mathbf{Q}\mathbf{z} + \mathbf{b}),$$

and thus

$$L_{i,j} = Q_{ii} + Q_{jj} - 2Q_{ij}. \tag{4}$$

## 4 Optimality Conditions and Measures

### 4.1 Conditions for Stationarity

We recall some well-known elementary concepts on optimality conditions for linearly constrained differentiable problems; for more details see, e.g., [6]. A vector $\mathbf{x}^* \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$ is called stationary if

$$\left\langle \nabla f\left(\mathbf{x}^*\right), \mathbf{x} - \mathbf{x}^*\right\rangle \geq 0 \quad \text{for all } \mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}.$$

If $\mathbf{x}^*$ is an optimal solution of (P), then it is also stationary. Therefore, stationarity is a necessary condition for optimality. When $f$ is in addition convex, then stationarity is a necessary and sufficient condition for optimality. Since the problem at hand is

linearly constrained, it follows that $\mathbf{x}^*$ is a stationary point if and only if the Karush-Kuhn-Tucker (KKT) conditions are satisfied, meaning that $\mathbf{x}^* \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$ is a stationary point of (P) if and only if there exists $\lambda \in \mathbb{R}$ for which

$$\nabla_i f(\mathbf{x}^*) = \begin{cases} = \lambda & l_i < x_i^* < u_i, \\ \leq \lambda & x_i^* = u_i, \\ \geq \lambda & x_i^* = l_i, \end{cases} \qquad i = 1, 2, \ldots, n.$$

The above characterization of optimality is stated in terms of a dual variable. It is also possible to rewrite the exact same condition solely in terms of the primal decision variables vector: $\mathbf{x}^*$ is a stationary point of (P) if and only if

$$\min_{j: x_j^* < u_j} \nabla_j f(\mathbf{x}^*) \geq \max_{i: x_i^* > l_i} \nabla_i f(\mathbf{x}^*). \tag{5}$$

*Remark 4.1* When the feasible set is the unit simplex ($K = 1, \mathbf{l} = \mathbf{0}, u_i = \infty, i = 1, \ldots, n$), condition (5) takes the form

$$\min_{j=1,\ldots,n} \nabla_j f(\mathbf{x}^*) \geq \max_{i: x_i^* > 0} \nabla_i f(\mathbf{x}^*), \tag{6}$$

and when there are no finite bounds, that is, when $l_i = -\infty$ and $u_i = \infty$ for all $i = 1, \ldots, n$, condition (5) takes the form

$$\min_{j=1,\ldots,n} \nabla_j f(\mathbf{x}^*) \geq \max_{i=1,\ldots,n} \nabla_i f(\mathbf{x}^*), \tag{7}$$

which of course just means that all the partial derivatives $\nabla_i f(\mathbf{x}^*)$ have the same value.

### 4.2 The Double-Sided Optimality Measure

The optimality condition (5) naturally calls for the following "optimality measure" (see also [39]):

$$R(\mathbf{x}) = \max\Big\{ \max_{i: x_i^* > l_i} \nabla_i f(\mathbf{x}^*) - \min_{j: x_j^* < u_j} \nabla_j f(\mathbf{x}^*), 0 \Big\}.$$

The above quantity is an optimality measure in the sense that it is positive for all nonstationary feasible points and zero for stationary points. However, it is actually a rather poor optimality measure because it is not even continuous, as the following example illustrates.

*Example 4.1* Consider the problem

$$\min\big\{(x_1 + 2x_2)^2 : x_1 + x_2 = 1, x_1, x_2 \geq 0\big\}. \tag{8}$$

The optimal solution of problem (8) is $\mathbf{x}^* = (1, 0)^T$ with corresponding gradient $\nabla f(\mathbf{x}^*) = (2, 4)^T$. The optimality condition is satisfied since $R(\mathbf{x}^*) =$

$\max\{\nabla_1 f(\mathbf{x}^*) - \nabla_2 f(\mathbf{x}^*), 0\} = \max\{-2, 0\} = 0$. Now, for any $\alpha \in (0, 0.5)$ consider the perturbed optimal solution

$$\mathbf{x}^*_\alpha = (1-\alpha, \alpha).$$

Obviously $\mathbf{x}^*_\alpha \to \mathbf{x}^*$ as $\alpha \to 0$. Since

$$\nabla f(\mathbf{x}^*_\alpha) = \begin{pmatrix} 2(1+\alpha) \\ 4(1+\alpha) \end{pmatrix},$$

it follows that

$$R\big(\mathbf{x}^*_\alpha\big) = \max\big\{4(1+\alpha) - 2(1+\alpha), 0\big\} = 2(1+\alpha),$$

which immediately implies that $R(\mathbf{x}^*_\alpha) \to 2$ $(\neq R(\mathbf{x}^*))$ as $\alpha \to 0$.

The discontinuity of $R(\cdot)$ is an evident drawback. Despite this, the optimality measure $R(\cdot)$ is essentially the basis of SMO methods for solving SVM training problems; see, e.g., [19, 20, 22, 25–28, 40], where at iteration $k$ the two indices chosen are related to those that cause the worst violation of optimality in terms of the value of $R$, namely,

$$\tilde{j} \in \operatorname*{argmin}_{j: x^k_j < u_j} \nabla_j f\big(\mathbf{x}^k\big), \qquad \tilde{i} \in \operatorname*{argmax}_{i: x^k_i > l_i} \nabla_i f\big(\mathbf{x}^k\big). \tag{9}$$

In this paper, we will consider the following optimality measure, which we will call *the double-sided optimality measure* (the subscript 2 and the "mysterious" square root will be clarified later on):

$$S_2(\mathbf{x}) = \max_{i \neq j} \left\{ \sqrt{\bar{L}_{i,j}} \min\left\{ \frac{1}{\bar{L}_{i,j}} \big[\nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x})\big], x_i - l_i, u_j - x_j \right\} \right\}, \tag{10}$$

where for any $i \neq j$, $\bar{L}_{i,j}$ is an upper bound on the local Lipschitz constant $L_{i,j}$. Note that, as opposed to $R(\mathbf{x})$, the double-sided optimality measure is continuous, which is a clear advantage.

We will now show two basic properties associated to $S_2(\cdot)$: (a) it is nonnegative and equal to zero only at stationary points, and (b) it can be computed by restricting the pairs of indices $(i, j)$ to those for which $\nabla_i f(\mathbf{x}) \geq \nabla_j f(\mathbf{x})$.

**Lemma 4.1** *For any $i \neq j$ let $\bar{L}_{i,j}$ be an upper bound on $L_{i,j}$. Then*

(a) *For any $\mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$ we have $S_2(\mathbf{x}) \geq 0$ and $S_2(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ is a stationary point of* (P).

(b) *For any $\mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$*

$$S_2(\mathbf{x}) = \max_{i \neq j: \nabla_i f(\mathbf{x}) \geq \nabla_j f(\mathbf{x})} \left\{ \sqrt{\bar{L}_{i,j}} \min\left\{ \frac{1}{\bar{L}_{i,j}} \big[\nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x})\big], x_i - l_i, u_j - x_j \right\} \right\}. \tag{11}$$

*Proof* (a) Let $i_0$, $j_0$ be two arbitrary different indices for which $\nabla_{i_0} f(\mathbf{x}) \geq \nabla_{j_0} f(\mathbf{x})$ (of course, such indices exist since for any $i \neq j$, either $(i, j)$ or $(j, i)$ satisfy the inequality between the partial derivatives). Then since $\mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$, we also have that $x_{i_0} \geq l_{i_0}$ and $x_{j_0} \leq u_{j_0}$, so that

$$\min\left\{\nabla_{i_0} f(\mathbf{x}) - \nabla_{j_0} f(\mathbf{x}), x_{i_0} - l_{i_0}, u_{j_0} - x_{j_0}\right\} \geq 0,$$

which immediately implies that $S_2(\mathbf{x}) \geq 0$. Therefore, $S_2(\mathbf{x}) = 0$ if and only if $S_2(\mathbf{x}) \leq 0$, which by the definition of $S_2$ is the same as the statement

$$\min\left\{\frac{1}{\bar{L}_{i,j}}\left[\nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x})\right], x_i - l_i, u_j - x_j\right\} \leq 0, \quad \text{for all } i \neq j.$$

If either $x_i = l_i$ or $x_j = u_j$, then the latter inequality is obvious. We can therefore rewrite the condition as:

$$\min\left\{\frac{1}{\bar{L}_{i,j}}\left[\nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x})\right], x_i - l_i, u_j - x_j\right\} \leq 0,$$
$$\text{for all } i : x_i > l_i \text{ and } j : x_j < u_j,$$

which is the same as

$$\nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x}) \leq 0, \quad \text{for all } i : x_i > l_i \text{ and } j : x_j < u_j,$$

that is,

$$\max_{i:x_i > l_i} \nabla_i f(\mathbf{x}) \leq \min_{j:x_j < u_j} \nabla_j f(\mathbf{x}),$$

meaning that $\mathbf{x}$ is a stationary point.

(b) If for some $i_0$, $j_0$ the inequality $\nabla_{i_0} f(\mathbf{x}) < \nabla_{j_0} f(\mathbf{x})$ is satisfied, then

$$\min\left\{\frac{1}{\bar{L}_{i_0,j_0}}\left[\nabla_{i_0} f(\mathbf{x}) - \nabla_{j_0} f(\mathbf{x})\right], x_{i_0} - l_{i_0}, u_{j_0} - x_{j_0}\right\} < 0. \tag{12}$$

Since $S_2(\mathbf{x}) \geq 0$, inequality (12) implies that the maximum in the definition (10) of the optimality measure $S_2(\cdot)$ is not attained at $(i_0, j_0)$, and therefore this pair of indices can be discarded in the maximization. The consequence is that the identity (11) is valid. □

### 4.3 The One-Sided Optimality Measure

Since one-sided simplex sets are special cases of two-sided simplex sets, we can also use $S_2(\cdot)$ as a measure for optimality for one-sided simplex sets. However, in this case we can also define a different optimality measure, which will be called the *one-sided measure* and is given by:

$$S_1(\mathbf{x}) = \max_{i=1,\ldots,n}\left\{\sqrt{\bar{L}_{i,J(\mathbf{x})}}\min\left\{\frac{1}{\bar{L}_{i,J(\mathbf{x})}}\left[\nabla_i f(\mathbf{x}) - \nabla_{J(\mathbf{x})} f(\mathbf{x})\right], x_i - l_i\right\}\right\}, \tag{13}$$

where here again $\bar{L}_{i,j}(i \neq j)$ is an upper bound on $L_{i,j}$. The index $J(\cdot)$ is defined by the relation

$$J(\mathbf{x}) \in \underset{j=1,2,\ldots,n}{\operatorname{argmin}} \nabla_j f(\mathbf{x}). \tag{14}$$

The measure $S_1$, similarly to $S_2$, is nonnegative and equal to zero only at stationary points, as the following lemma states.

**Lemma 4.2** *Consider the one-sided simplex set $\Delta_n^{K,\mathbf{l}}$. For any $\mathbf{x} \in \Delta_n^{K,\mathbf{l}}$ the inequality $S_1(\mathbf{x}) \geq 0$ is satisfied and $S_1(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ is a stationary point of* (P).

*Proof* By the definition of $J$ (relation (14)), it follows that $\nabla_i f(\mathbf{x}) \geq \nabla_{J(\mathbf{x})} f(\mathbf{x})$ for any $i \in \{1, 2, \ldots, n\}$, which readily establishes the nonnegativity of $S_1$. Therefore, for any $\mathbf{x} \in \Delta_n^{K,\mathbf{l}}$, we have $S_1(\mathbf{x}) = 0$ if and only if $S_1(\mathbf{x}) \leq 0$, which is equivalent to the relation

$$\nabla_i f(\mathbf{x}) \leq \nabla_{J(\mathbf{x})} f(\mathbf{x})$$

for all $i$ satisfying $x_i > l_i$. That is, $S_1(\mathbf{x}) = 0$ if and only if

$$\max_{i:x_i > l_i} \nabla_i f(\mathbf{x}) \leq \min_{j=1,2,\ldots,n} \nabla_j f(\mathbf{x}),$$

meaning that $\mathbf{x}$ is a stationary point of (P). □

Note that despite the fact that the optimal $i$ was not chosen a priori in the definition of $S_1$ to be different from $J(\mathbf{x})$, the optimal $i$ will be different than $J(\mathbf{x})$ whenever $\mathbf{x}$ is not a stationary point.

### 4.4 $S_2$ Versus $S_1$

From now on, we will assume that there are two possible settings which correspond to a parameter $M$ that takes the values 1 and 2.

**The Two Settings**:

- One-sided setting ($M = 1$): The feasible set is $\Delta_n^{K,\mathbf{l}}$ ($u_i = \infty \; \forall i$)
- Two-sided setting ($M = 2$): The feasible set is $\Delta_n^{K,\mathbf{l},\mathbf{u}}$.

The measure $S_2$ is also relevant in the one-sided case, in which it takes the form:

$$\max_{i \neq j : \nabla_i f(\mathbf{x}) \geq \nabla_j f(\mathbf{x})} \left\{ \sqrt{\bar{L}_{i,j}} \min \left\{ \frac{1}{\bar{L}_{i,j}} \left[ \nabla_i f(\mathbf{x}) - \nabla_j f(\mathbf{x}) \right], x_i - l_i \right\} \right\}.$$

However, in the one-sided case we will be more interested in the measure $S_1$ for two reasons. First of all, it is easier to compute—it requires only $O(n)$ computations and not $O(n^2)$ as is required by $S_2$. In addition, as will be clarified later on, some results on the one-sided setting can be obtained only when exploiting the measure $S_1$ rather than $S_2$ (cf. Sect. 6). At this point, we also note that the idea of reducing the cost of indices selection from $O(n^2)$ to $O(n)$ by choosing $j_k = J(\mathbf{x}^k)$ can be traced back to [29].

## 5 The 2-Coordinate Descent Method for Solving (P)

### 5.1 Description of the Method

Motivated by the definition of the new optimality measures $S_1(\cdot)$ and $S_2(\cdot)$, we now define a schematic coordinate descent method where at each iteration a descent step is performed with respect to only two variables while keeping all other variables fixed.

**The 2-Coordinate Descent Method**

**Input**: $\bar{L}_{i,j}$—an upper bound on $L_{i,j}$ $(i, j \in \{1, 2, \ldots, n\}, i \neq j)$

**Initialization**: $\mathbf{x}^0 \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$.
**General Step** $(k = 0, 1, \ldots)$:

(a) Choose two different indices $(i_k, j_k)$ for which $\nabla_{i_k} f(\mathbf{x}^k) - \nabla_{j_k} f(\mathbf{x}^k) \geq 0$.

(b) Set

$$\mathbf{x}^{k+1} = \mathbf{x}^k + T_k(\mathbf{e}_{i_k} - \mathbf{e}_{j_k}),$$

where $T_k \in [\max\{l_{i_k} - x_{i_k}^k, x_{j_k}^k - u_{j_k}\}, 0]$.

There are two important details that are missing in the above description. First, the index selection strategy of $(i_k, j_k)$ was not given, and second, the choice of stepsize $T_k$ should be made precise. The index selection strategy depends on the specific setting (i.e., $M = 1$ or $M = 2$), and the two chosen indices are those that cause the largest violation of optimality in terms of the optimality measures $S_1$ and $S_2$:

– **Double-sided index selection strategy** $(M = 2)$:

$$(i_k, j_k) \in \underset{i \neq j : \nabla_i f(\mathbf{x}) \geq \nabla_j f(\mathbf{x})}{\operatorname{argmax}} \left\{ \sqrt{\bar{L}_{i,j}} \min\left\{ \frac{1}{\bar{L}_{i,j}} \left[\nabla_i f\left(\mathbf{x}^k\right) - \nabla_j f\left(\mathbf{x}^k\right)\right], x_i^k - l_i, u_j - x_j^k \right\} \right\}. \tag{15}$$

– **One-sided index selection strategy** $(M = 1)$:

$$j_k = J\left(\mathbf{x}^k\right) \in \underset{j=1,\ldots,n}{\operatorname{argmin}} \nabla_j f\left(\mathbf{x}^k\right), \tag{16}$$

$$i_k \in \underset{i=1,\ldots,n}{\operatorname{argmax}} \left\{ \sqrt{\bar{L}_{i,j_k}} \min\left\{ \frac{1}{\bar{L}_{i,j_k}} \left[\nabla_i f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right], x_i^k - l_i \right\} \right\}. \tag{17}$$

As for the stepsize $T_k$, note that since $\nabla_{i_k} f(\mathbf{x}^k) - \nabla_{j_k} f(\mathbf{x}^k) \geq 0$, it follows that $\phi'_{i_k,j_k,\mathbf{x}^k}(0) \geq 0$, which means that the directional derivative of $f$ at $\mathbf{x}^k$ in the direction $\mathbf{e}_{i_k} - \mathbf{e}_{j_k}$ is nonnegative. This enforces (to ensure the nonincreasing property of the objective function values sequence) that $T_k$ is nonpositive. In addition, in order to guarantee feasibility of the next iterate, we also assume that $T_k \in I_{i_k,j_k,\mathbf{x}^k}$, which combined with the nonpositivity of $T_k$ implies that $T_k$ resides in the interval $[\max\{l_{i_k} - x_{i_k}^k, x_{j_k}^k - u_{j_k}\}, 0]$. Two specific choices of stepsize selection methods that we will consider are:

– **Full Minimization Step**. In this case $T_k$ is chosen to minimize the objective function in the direction $\mathbf{d} = -(\mathbf{e}_{i_k} - \mathbf{e}_{j_k})$:

$$T_k \in \operatorname*{argmin}_t \left\{ f\left(\mathbf{x}^k + t(\mathbf{e}_{i_k} - \mathbf{e}_{j_k})\right) : t \in \left[\max\left\{l_{i_k} - x^k_{i_k}, x^k_{j_k} - u_{j_k}\right\}, 0\right]\right\}.$$

– **Partial Minimization Step**. In this case $T_k$ is chosen as

$$T_k = \max\left\{ -\frac{1}{\bar{L}_{i_k,j_k}}\left(\nabla_{i_k} f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right), l_{i_k} - x^k_{i_k}, x^k_{j_k} - u_{j_k}\right\}. \tag{18}$$

In general, given the $k$-th iterate $\mathbf{x}^k$ and the gradient $\nabla f(\mathbf{x}^k)$, the determination of the two indices $i_k$ and $j_k$ in the double-sided setting requires $O(n^2)$ computations. This is in contrast to the one-sided setting in which only $O(n)$ computations are required. A reduction of computations when $M = 2$ can be made by noting that we can restrict ourselves only to indices $(i, j)$ for which $x_i > l_i$ and $x_j < u_j$:

$$(i_k, j_k) \in \operatorname*{argmax}_{i \neq j : \nabla_i f(\mathbf{x}) \geq \nabla_j f(\mathbf{x}), x^k_i > l_i, x^k_j < u_j} \left\{ \sqrt{\bar{L}_{i,j}} \min\left\{ \frac{1}{\bar{L}_{i,j}}\left[\nabla_i f\left(\mathbf{x}^k\right) - \nabla_j f\left(\mathbf{x}^k\right)\right], x^k_i - l_i, u_j - x^k_j \right\}\right\}.$$

Therefore, it is enough to perform $O(pr)$ computations, where $p = \#\{i : x^k_i > l_i\}$ and $r = \#\{j : x^k_j < u_j\}$. This observation is the basis of a significant reduction in the amount of computations required to find the two indices $(i_k, j_k)$, for example when the lower bounds are all zero ($l_i = 0$) and the iterates are sparse. This is the typical situation in the SVM problem which will be described in Sect. 7.3.

Since we deal with two different settings ($M = 1$ and $M = 2$), and each setting has two possibilities for the stepsize selection strategy (full or partial), it follows that we actually consider four different algorithms. However, the basic convergence results for the four algorithms are derived in a unified manner.

### 5.2 Convergence

We begin the convergence analysis by first recalling the following fundamental result which is frequently used in order to establish convergence of gradient-based methods [41, 42].

**Lemma 5.1** *Consider the problem*

$$\min\left\{g(\mathbf{x}) : \mathbf{x} \in X\right\},$$

*where $X \subseteq \mathbb{R}^d$ is a closed and convex set and $g : \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable with a Lipschitz gradient. Let $L_g$ be an upper bound on the Lipschitz constant of $\nabla g$. Then*

$$g(\mathbf{x}) - g\left(P_X\left[\mathbf{x} - \frac{1}{L_g}\nabla g(\mathbf{x})\right]\right) \geq \frac{L_g}{2}\left\|\mathbf{x} - P_X\left[\mathbf{x} - \frac{1}{L_g}\nabla g(\mathbf{x})\right]\right\|^2 \tag{19}$$

*for any $\mathbf{x} \in X$.*

Relying on Lemma 5.1, we will now show that the 2-coordinate descent method is guaranteed to achieve at each iteration a decrease of the objective function which is at least half of the squared optimality measure.

**Theorem 5.1** *Let $\{\mathbf{x}^k\}$ be the sequence generated by the 2-coordinate descent method with either full or partial stepsize selection strategies. Then the following relation holds for every $k = 0, 1, \ldots$:*

$$f\left(\mathbf{x}^k\right) - f\left(\mathbf{x}^{k+1}\right) \geq \frac{1}{2} S_M^2\left(\mathbf{x}^k\right). \tag{20}$$

*Proof* The partial minimization stepsize now denoted by $T_k^p$ is given by (see (18)):

$$T_k^p = -\min\left\{\frac{1}{\bar{L}_{i_k,j_k}}\left(\nabla_{i_k} f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right), x_{i_k}^k - l_{i_k}, u_{j_k} - x_{j_k}^k\right\}.$$

Substituting $X = [\max\{l_{i_j} - x_{i_k}^k, x_{j_k} - u_{j_k}\}, 0]$, $g := \phi_{i_k,j_k,\mathbf{x}^k}$, $\mathbf{x} = 0$ and $L_g = \bar{L}_{i_k,j_k}$ into (19), we obtain that

$$\begin{aligned}
&\phi_{i_k,j_k,\mathbf{x}^k}(0) - \phi_{i_k,j_k,\mathbf{x}^k}\left(0 - P_X\left(0 - \frac{1}{\bar{L}_{i_k,j_k}}\phi'_{i_k,j_k,\mathbf{x}^k}(0)\right)\right) \\
&\quad \geq \frac{\bar{L}_{i_k,j_k}}{2}\left|0 - P_X\left(0 - \frac{1}{\bar{L}_{i_k,j_k}}\phi'_{i_k,j_k,\mathbf{x}^k}(0)\right)\right|^2.
\end{aligned} \tag{21}$$

Recall that by (3)

$$\phi'_{i_k,j_k,\mathbf{x}^k}(0) = \nabla_{i_k} f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right) \geq 0, \tag{22}$$

$$\phi_{i_k,j_k,\mathbf{x}^k}(0) = f\left(\mathbf{x}^k\right). \tag{23}$$

Also, since $\phi'_{i_k,j_k,\mathbf{x}^k}(0) \geq 0$, it follows that

$$\begin{aligned}
&P_X\left(0 - \frac{1}{\bar{L}_{i_k,j_k}}\phi'_{i_k,j_k,\mathbf{x}^k}(0)\right) \\
&\quad = \max\left\{-\frac{1}{\bar{L}_{i_k,j_k}}\phi'_{i_k,j_k,\mathbf{x}^k}(0), \max\left\{l_{i_j} - x_{i_k}^k, x_{j_k} - u_{j_k}\right\}\right\} \\
&\quad = -\min\left\{\frac{1}{\bar{L}_{i_k,j_k}}\left(\nabla_{i_k} f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right), x_{i_k}^k - l_{i_k}, u_{j_k} - x_{j_k}^k\right\} \\
&\quad = T_k^p.
\end{aligned} \tag{24}$$

Using (22), (23), and (24), the inequality (21) becomes

$$\begin{aligned}
&\phi_{i_k,j_k,\mathbf{x}^k}(0) - \phi_{i_k,j_k,\mathbf{x}^k}\left(T_k^p\right) \\
&\quad \geq \frac{\bar{L}_{i_k,j_k}}{2}\min\left\{\frac{1}{\bar{L}_{i_k,j_k}}\left(\nabla_{i_k} f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right), x_{i_k}^k - l_{i_k}, u_{j_k} - x_{j_k}^k\right\}^2 \\
&\quad = \frac{1}{2} S_M^2\left(\mathbf{x}^k\right),
\end{aligned}$$

where the last equality follows from the relations (15), (16), and (17) defining $(i_k, j_k)$. Therefore,

$$f(\mathbf{x}^k) - f(\mathbf{x}^k + T_k^p(\mathbf{e}_{i_k} - \mathbf{e}_{j_k})) \geq \frac{1}{2} S_M^2(\mathbf{x}^k), \tag{25}$$

which is exactly inequality (20) for the partial minimization stepsize case. When the stepsize is chosen via the full minimization strategy, (20) follows by combining the obvious relation $f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k + T_k^p(\mathbf{e}_{i_k} - \mathbf{e}_{j_k}))$ with (25). □

We can now establish the main convergence result of the 2-coordinate descent method in the nonconvex case.

**Theorem 5.2** *Let $\{\mathbf{x}^k\}$ be the sequence generated by the 2-coordinate descent method with either full or partial minimization stepsize. Then*

a. $S_M(\mathbf{x}^k) \to 0$ *as* $k \to \infty$.
b. *For every* $n = 0, 1, 2, \ldots,$

$$\min_{k=0,1,\ldots,n} S_M(\mathbf{x}^k) \leq \sqrt{2(f(\mathbf{x}^0) - f^*)} \frac{1}{\sqrt{n+1}}. \tag{26}$$

c. *Any accumulation point of the sequence $\{\mathbf{x}^k\}$ is a stationary point.*
d. *If the objective function is convex and the level set*

$$\mathcal{L}(f, f(\mathbf{x}^0)) = \{\mathbf{x} \in \Delta_n^{K,\mathbf{l},\mathbf{u}} : f(\mathbf{x}) \leq f(\mathbf{x}^0)\}$$

*is bounded, then $f(\mathbf{x}^k) \to f^*$ where $f^*$ is the optimal value of problem* (P).

*Proof* a. The sequence $\{f(\mathbf{x}^k)\}$ is nonincreasing and bounded below and thus converges. Therefore, by inequality (20), it follows that $S_M(\mathbf{x}^k) \to 0$ as $k \to \infty$.

b. Summing inequality (20) over $k = 0, 1, \ldots, n$, we obtain that

$$\sum_{k=0}^{n} (f(\mathbf{x}^k) - f(\mathbf{x}^{k+1})) \geq \frac{1}{2} \sum_{k=0}^{n} S_M^2(\mathbf{x}^k) \geq \frac{n+1}{2} \min_{k=0,1,\ldots,n} S_M^2(\mathbf{x}^k).$$

Thus,

$$f(\mathbf{x}^0) - f(\mathbf{x}^{n+1}) \geq \frac{n+1}{2} \min_{k=0,1,\ldots,n} S_M^2(\mathbf{x}^k),$$

which combined with the inequality $f(\mathbf{x}^{n+1}) \geq f^*$, establishes the desired result (26).

c. By part a, and the continuity of the optimality measure $S_M(\cdot)$, it follows that if $\tilde{\mathbf{x}}$ is an accumulation point of the sequence, then $S_M(\tilde{\mathbf{x}}) = 0$, which implies by Lemma 4.2 and by part a of Lemma 4.1 that $\tilde{\mathbf{x}}$ is a stationary point of (P).

d. Since the sequence $\{f(\mathbf{x}^k)\}$ is nonincreasing, it follows that it is bounded as it is contained in the bounded level set $\mathcal{L}(f, f(\mathbf{x}^0))$. Therefore, it has a subsequence $\{\mathbf{x}^{k_n}\}$ that converges to an accumulation point $\mathbf{x}^*$, which by part c is a stationary point. The convexity of $f$ implies that $\mathbf{x}^*$ is an optimal solution of (P). Finally, the continuity of $f$ yields that $f(\mathbf{x}^*) = f^*$, meaning that $\{f(\mathbf{x}^k)\}$ converges to the optimal value of the problem. □

*Remark 5.1* Asymptotic results such as the one established in part c of Theorem 5.2 were obtained for other decomposition methods designed to solve the general problem (P) in the works [24, 25, 27–31]; see the introduction for further details. The Lipschitz continuity of the gradient is not assumed in these works, and the key property used in the latter works is that the objective function is continuously differentiable.

## 6 The Convex Case: Rate of Convergence in the One-Sided Setting

In this section, we consider the one-sided setting ($M = 1$) in the case when the objective function is convex and the lower bounds are finite ($l_i > -\infty$ for all $i = 1, 2, \ldots, n$). In this case the feasible set $\Delta_n^{K,\mathbf{l}}$ is bounded, and the following soon-to-be useful notation is well defined:

$$B = \max_{\mathbf{x} \in \Delta_n^{K,\mathbf{l}}} \|\nabla f(\mathbf{x})\|_\infty. \tag{27}$$

Our main objective is to establish a nonasymptotic sublinear rate of convergence of the sequence of function values. We begin with the following useful lemma.

**Lemma 6.1** *Suppose that $f$ is convex and $M = 1$. Let $\{\mathbf{x}^k\}$ be the sequence generated by the 2-coordinate descent method with either full or partial stepsize selection strategies. Then*

$$f\left(\mathbf{x}^k\right) - f^* \leq A(n-1)S_1\left(\mathbf{x}^k\right), \tag{28}$$

*where*

$$A = \max\left\{\sqrt{\bar{L}_{\max}}R, \frac{2B}{\sqrt{\bar{L}_{\min}}}\right\}, \tag{29}$$

*and $R, \bar{L}_{\min}, \bar{L}_{\max}$ are defined by*

$$\bar{L}_{\max} = \max_{i \neq j} \bar{L}_{ij}, \qquad \bar{L}_{\min} = \min_{i \neq j} \bar{L}_{ij}, \tag{30}$$

$$R = K - \min_{i=1,\ldots,n} \{l_i\}. \tag{31}$$

*Proof* To simplify the presentation of the proof, we will assume without loss of generality that $j_k = J(\mathbf{x}^k) = n$, so that

$$S_1(\mathbf{x}) = \max_{i=1,\ldots,n} \left\{\sqrt{\bar{L}_{i,n}} \min\left\{\frac{1}{\bar{L}_{i,n}}\left[\nabla_i f(\mathbf{x}) - \nabla_n f(\mathbf{x})\right], x_i - l_i\right\}\right\}.$$

Consider the $n \times (n-1)$ matrix defined by

$$\mathbf{L} = \begin{pmatrix} \mathbf{I}_{n-1} \\ -\mathbf{e}^T \end{pmatrix}.$$

By the definition of $\mathbf{L}$ we have that

$$\Delta_n^{K,\mathbf{l}} = \left\{\mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda} : \boldsymbol{\lambda} \in \mathbb{R}^{n-1}, \mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda} \geq \mathbf{l}\right\}. \tag{32}$$

Consequently, problem (P) can be rewritten as

$$\text{(P')} \quad \min\{g(\boldsymbol{\lambda}) : \boldsymbol{\lambda} \in \mathbb{R}^{n-1}, \mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda} \geq \mathbf{l}\},$$

where

$$g(\boldsymbol{\lambda}) := f\left(\mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda}\right) \quad \text{for all } \boldsymbol{\lambda} : \mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda} \geq \mathbf{l}.$$

Let $\mathbf{x}^*$ be an optimal solution of (P). By (32) it follows that there exists $\boldsymbol{\lambda}^* \in \mathbb{R}^{n-1}$ satisfying $\mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda}^* \geq \mathbf{l}$ for which $\mathbf{x}^* = \mathbf{x}^k + \mathbf{L}\boldsymbol{\lambda}^*$. We can therefore write:

$$f\left(\mathbf{x}^k\right) - f^* = f\left(\mathbf{x}^k\right) - f\left(\mathbf{x}^*\right) = g(\mathbf{0}) - g\left(\boldsymbol{\lambda}^*\right) \leq \nabla g(\mathbf{0})^T\left(\mathbf{0} - \boldsymbol{\lambda}^*\right) = -\nabla g(\mathbf{0})^T\boldsymbol{\lambda}^*, \tag{33}$$

where the inequality follows from the convexity of $g$. Note that from the definition of $\mathbf{L}$ we have

$$\lambda_i^* = x_i^* - x_i^k, \quad i = 1, 2, \ldots, n-1. \tag{34}$$

In addition,

$$\nabla g(\mathbf{0}) = \mathbf{L}^T \nabla f\left(\mathbf{x}^k\right) = \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)_{i=1}^{n-1}. \tag{35}$$

Substituting (34) and (35) into (33), we obtain that

$$f\left(\mathbf{x}^k\right) - f^* \leq \sum_{i=1}^{n-1}\left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - x_i^*\right). \tag{36}$$

For every $i \in \{1, 2, \ldots, n-1\}$ satisfying $x_i^k = l_i$ we have (recalling that $\nabla_i f(\mathbf{x}^k) \geq \nabla_n f(\mathbf{x}^k)$):

$$\left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - x_i^*\right) = \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(l_i - x_i^*\right) \leq 0,$$

which combined with (36) implies that

$$\begin{aligned} f\left(\mathbf{x}^k\right) - f^* &\leq \sum_{i\in\{1,\ldots,n-1\}:x_i^k > l_i} \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - x_i^*\right) \\ &\leq \sum_{i\in\{1,\ldots,n-1\}:x_i^k > l_i} \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - l_i\right). \end{aligned} \tag{37}$$

Now, by using the definition of $B$ (27) we have:

$$\begin{aligned} \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - l_i\right) &\leq 2B\left(x_i^k - l_i\right), \\ \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)\left(x_i^k - l_i\right) &\leq \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right) \max_{\mathbf{x}\in\Delta_n^{K,\mathbf{l}}} \|\mathbf{x} - \mathbf{l}\|_\infty \\ &= \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_n f\left(\mathbf{x}^k\right)\right)R, \end{aligned}$$

where $R$ is defined in (31). We thus have for any $i = 1, 2, \ldots, n-1$:

$$
\begin{aligned}
&\big(\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big)\big(x_i^k - l_i\big)\\
&\quad\le \min\big\{\big(\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big)R, 2\big(x_i^k - l_i\big)B\big\}\\
&\quad\le \min\left\{R\bar{L}_{\mathrm{in}}\frac{1}{\bar{L}_{\mathrm{in}}}\big[\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big], 2\big(x_i^k - l_i\big)B\right\}\\
&\quad\le \max\{R\bar{L}_{\mathrm{in}}, 2B\}\cdot\min\left\{\frac{1}{\bar{L}_{\mathrm{in}}}\big[\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big], x_i^k - l_i\right\}\\
&\quad\le \max\left\{R\sqrt{\bar{L}_{\mathrm{in}}}, \frac{2B}{\sqrt{\bar{L}_{\mathrm{in}}}}\right\}\sqrt{\bar{L}_{\mathrm{in}}}\cdot\min\left\{\frac{1}{\bar{L}_{\mathrm{in}}}\big[\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big], x_i^k - l_i\right\}\\
&\quad\le A\sqrt{\bar{L}_{\mathrm{in}}}\cdot\min\left\{\frac{1}{\bar{L}_{\mathrm{in}}}\big[\nabla_i f\big(\mathbf{x}^k\big) - \nabla_n f\big(\mathbf{x}^k\big)\big], x_i^k - l_i\right\}\\
&\quad\le AS_1\big(\mathbf{x}^k\big),
\end{aligned}
$$

where $A$ is defined in (29). We can therefore conclude from (37) that

$$f\big(\mathbf{x}^k\big) - f^* \le A(n-1)S_1\big(\mathbf{x}^k\big),$$

which is the desired result. □

To establish the $O(1/k)$ rate of convergence, we will use the following simple and well-known lemma on sequences of nonnegative numbers (for a proof, see for example [43]):

**Lemma 6.2** *Let $\{a_k\}_{k\ge 0}$ be a sequence of nonincreasing and nonnegative numbers satisfying*

$$a_k - a_{k+1} \ge \gamma a_k^2, \quad k = 0, 1, 2, \ldots$$

*for some positive number $\gamma$. Then*

$$a_k \le \frac{1}{\gamma k}, \quad k = 1, 2, \ldots$$

We are now ready to prove our main result.

**Theorem 6.1** *Suppose that $f$ is convex and $M = 1$. Let $\{\mathbf{x}^k\}$ be the sequence generated by the 2-coordinate descent method. Then*

$$f\big(\mathbf{x}^k\big) - f^* \le \frac{2A^2(n-1)^2}{k},$$

*where $A$ is given in (29).*

*Proof* By combining (20) and (28) we have

$$f\big(\mathbf{x}^k\big) - f\big(\mathbf{x}^{k+1}\big) \ge \frac{1}{2}S_1^2\big(\mathbf{x}^k\big) \ge \frac{1}{2A^2(n-1)^2}\big(f\big(\mathbf{x}^k\big) - f^*\big)^2.$$

Therefore,

$$\big(f\big(\mathbf{x}^k\big)-f^*\big)-\big(f\big(\mathbf{x}^{k+1}\big)-f^*\big)\geq\frac{1}{2A^2(n-1)^2}\big(f\big(\mathbf{x}^k\big)-f^*\big)^2.$$

Invoking Lemma 6.2 with $a_k:=f(\mathbf{x}^k)-f^*$ and $\gamma=\frac{1}{2A^2(n-1)^2}$, the desired result follows. □

*Remark 6.1* It is not difficult to see that the proof of Lemma 6.1 can be refined and the inequality (28) can be replaced with

$$f\big(\mathbf{x}^k\big)-f^*\leq A\big(\|\mathbf{x}^k-\mathbf{l}\|_0-1\big)S_1\big(\mathbf{x}^k\big),$$

where for a vector $\mathbf{y}$, $\|\mathbf{y}\|_0$ stands for the number of nonzero elements in $\mathbf{y}$. Therefore, if for example the feasible set is the unit simplex and the sparsity of all the iterates is bounded via $\|\mathbf{x}^k\|_0\leq p$, then $f(\mathbf{x}^k)-f^*\leq A(p-1)S_1(\mathbf{x}^k)$, and the complexity result will be replaced by

$$f\big(\mathbf{x}^k\big)-f^*\leq\frac{2A^2(p-1)^2}{k},$$

which is a significant improvement when $p\ll n$. This sparsity property is rather common in several applications such as the Chebyshev center problem, which will be described in Sect. 7.1.

*Remark 6.2* We note that for the dual SVM problem, an asymptotic linear rate of convergence was established in [44] under the assumption that $\mathbf{Q}$ is positive definite and that strict complementarity holds at the optimal primal-dual solution. This result was also used in order to show the asymptotic linear rate of convergence of the decomposition method derived in [29], which exploits second order information in the index selection strategy. Our objective here was to derive a *nonasymptotic* rate of convergence result, i.e., establish the fact that the accuracy of an iterative optimization algorithm can be guaranteed to hold from the first iteration and not only for a large enough value of the iteration counter $k$.

### 6.1 Unknown Lipschitz Constants

When the Lipschitz constants are not known, the 2-coordinate descent method—as described—cannot be employed. Specifically, the Lipschitz constants have two roles: first, they are used in the index selection step, and second, when a partial minimization step is employed, the knowledge of the corresponding Lipschitz constant is required. However, as will be described now, it is possible to adjust the algorithm for the case when the Lipschitz constants are not known by incorporating a backtracking procedure for finding "estimates" of the Lipschitz constants. Specifically, suppose that the initial estimates of the local Lipschitz constants are given by $\bar{L}_{ij}^{(-1)}$ (for all $i\neq j$). For any $k\geq 1$ the local Lipschitz constant estimates at iteration $k$, which are denoted by $\bar{L}_{ij}^{(k)}$, are generated from the estimates of the previous iteration $\bar{L}_{ij}^{(k-1)}$ by the following backtracking procedure:

**Procedure LCE: Lipschitz Constants Estimates Generation**:

**Input**: a constant $\eta > 1$.

– $j_k = J(\mathbf{x}^k) \in \operatorname{argmin}_{j=1,\ldots,n} \nabla_j f(\mathbf{x}^k)$.
– For any $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, n\}/\{j_k, i\}$:

$$\bar{L}_{i,j}^{(k)} = \bar{L}_{i,j}^{(k-1)}$$

– For any $i \neq j_k$ for which $x_i = l_i$:

$$\bar{L}_{i,j_k}^{(k)} = \bar{L}_{i,j_k}^{(k-1)}$$

– For any $i \neq j_k$ for which $x_i > l_i$, let $i_k$ be the smallest nonnegative integer such that with $\tilde{L}_k = \eta^{i_k} \bar{L}_{i,j_k}^{(k-1)}$ the inequality

$$f\left(\mathbf{x}^k + \tilde{T}_{i,j_k,k}^{\tilde{L}_k}(\mathbf{e}_i - \mathbf{e}_{j_k})\right) \leq f\left(\mathbf{x}^k\right) - \left(\nabla_i f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right)\tilde{T}_{i,j_k,k}^{\tilde{L}_k} + \frac{\tilde{L}_k}{2}\left(\tilde{T}_{i,j_k,k}^{\tilde{L}_k}\right)^2$$

is satisfied with $\tilde{T}_{i,j,k}^{M} = \max\{-\frac{1}{M}(\nabla_i f(\mathbf{x}^k) - \nabla_j f(\mathbf{x}^k)), l_i - x_i^k\}$.
Set $\bar{L}_{i,j_k}^{(k)} = \eta^{i_k} \bar{L}_{i,j_k}^{(k-1)}$.

Given the above procedure for the estimation of the Lipschitz constants, it is not difficult to define a method that does not require the explicit knowledge of these constants.

**The 2-Coordinate Descent Method with Backtracking**

**Input**: $\bar{L}_{i,j}^{(-1)} (i \neq j)$- initial estimates on the local Lipschitz constants.

**Initialization**: $\mathbf{x}^0 \in \Delta_n^{K,\mathbf{l},\mathbf{u}}$.
**General Step** ($k = 0, 1, \ldots$):

(a) Set $j_k = J(\mathbf{x}^k) \in \operatorname{argmin}_{j=1,\ldots,n} \nabla_j f(\mathbf{x}^k)$.
(b) Compute the local Lipschitz constant estimates $\bar{L}_{i,j}^{(k)}$ according to procedure LCE.
(c) Define

$$i_k \in \operatorname*{argmax}_{i \neq j_k} \left\{ \sqrt{\bar{L}_{i,j_k}^{(k)}} \min\left\{ \frac{1}{\bar{L}_{i,j}^{(k)}} \left[\nabla_i f\left(\mathbf{x}^k\right) - \nabla_{j_k} f\left(\mathbf{x}^k\right)\right], x_i^k - l_i \right\} \right\}.$$

(d) Set

$$\mathbf{x}^{k+1} = \mathbf{x}^k + T_k(\mathbf{e}_{i_k} - \mathbf{e}_{j_k}),$$

where $T_k$ is either computed via a full minimization strategy or via the formula $T_k = \max\{-\frac{1}{\bar{L}_{i_k,j_k}^{(k)}}(\nabla_{i_k} f(\mathbf{x}^k) - \nabla_{j_k} f(\mathbf{x}^k)), l_{i_k} - x_{i_k}^k\}$

The analysis of the above method is very technical, and is based on the analysis employed in the known Lipschitz constants case. We will therefore state the main convergence result without a proof.

**Theorem 6.2** *Suppose that $f$ is convex and $M = 1$. Let $\{\mathbf{x}^k\}$ be the sequence generated by the 2-coordinate descent method with backtracking. Then*

$$f\left(\mathbf{x}^k\right) - f^* \leq \max\left\{\sqrt{\eta L_{\max}}R, \frac{2B}{\sqrt{\bar{L}_{\min}^{(-1)}}}\right\}^2 (n-1)^2 \frac{1}{2k}, \quad k = 1, 2, \ldots$$

*where*

$$L_{\max} = \max_{i \neq j} L_{i,j},$$
$$\bar{L}_{\min}^{(-1)} = \min_{i \neq j} \bar{L}_{i,j}^{(-1)}.$$

## 7 Examples and Numerical Illustrations

In order to demonstrate the potential of the 2-coordinate descent methods described in the paper, we give two illustrative examples and report several experiments on some SVM training problems. All the experiments are performed on the class of quadratic convex objective functions for which the Lipschitz constant can be easily obtained from the maximum eigenvalue of the Hessian matrix. We use the acronym 2cd for the 2-coordinate descent method.

### 7.1 A Chebyshev Center Example

Given a set of points $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m \in \mathbb{R}^d$, a known geometrical problem is to find their *Chebyshev center*, which is the center of the minimum radius ball enclosing all the points. There exist many algorithms for solving the Chebyshev center problem; see for example the paper [38], which also contains an overview of the problem as well as many relevant references. Mathematically, the problem can be directly formulated as the following convex optimization problem ($r$ stands for the squared radius and $\mathbf{x}$ is the Chebyshev center):

$$\min_{r \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^d} \left\{r : \|\mathbf{x} - \mathbf{a}_i\|^2 \leq r, i = 1, \ldots, m\right\}, \tag{38}$$

which is of course *not* of the form (P). However, a standard computation shows that the dual of (38) is of the form of problem (P) with a quadratic objective and a unit simplex as the feasible set:

$$\max\left\{-\|\mathbf{A}\boldsymbol{\lambda}\|^2 + \sum_{i=1}^{m} \|\mathbf{a}_i\|^2 \lambda_i : \boldsymbol{\lambda} \in \Delta_m\right\}, \tag{39}$$

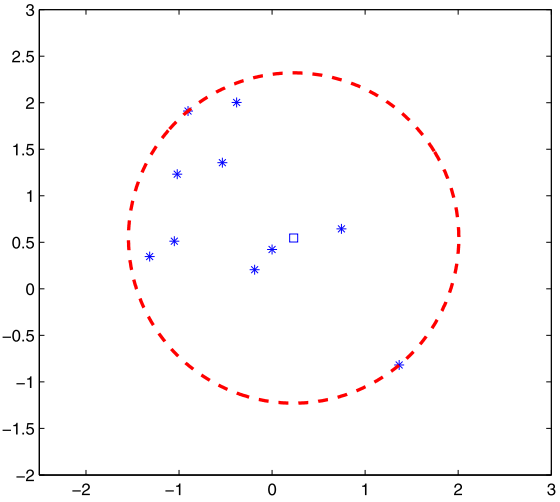


**Fig. 1** Ten points in the plane and their Chebyshev center (denoted by a *square*) and minimum-radius inscribing *circle*

where the columns of $\mathbf{A} \in \mathbb{R}^{d\times n}$ are the $m$ vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$. Given an optimal solution $\boldsymbol{\lambda}^*$ of the dual problem (39), the Chebyshev center is given by $\mathbf{x}_{\text{che}} = \mathbf{A}\boldsymbol{\lambda}^*$ and the radius of the minimum enclosing ball is the square root of the optimal objective function $\sqrt{-\|\mathbf{A}\boldsymbol{\lambda}^*\|^2 + \sum_{i=1}^m \|\mathbf{a}_i\|^2\lambda_i^*}$. The solution of the dual problem (39) tends to be sparse since the nonzero components of the optimal solution correspond to points which are on the boundary of the optimal ball, and usually there are only a few such points. For example, in Fig. 1 the optimal solution of the Chebyshev center problem with $d = 2$ and $m = 10$ is given (the center is denoted by a square). Note that there are only two points on the boundary of the circle, and these points correspond to the only two nonzero components of the optimal solution $\boldsymbol{\lambda}^*$, which is given by $\boldsymbol{\lambda} = (0, 0, 0, 0, 0, 0.5, 0.5, 0, 0, 0)^T$.

The dual problem can also be formulated as a minimization problem:

$$\min\left\{q(\boldsymbol{\lambda}) := \|\mathbf{A}\boldsymbol{\lambda}\|^2 - \sum_{i=1}^m \|\mathbf{a}_i\|^2\lambda_i : \boldsymbol{\lambda} \in \Delta_m\right\}. \tag{40}$$

In this set of runs we generated 2,000 points in $\mathbb{R}^2$, where all the components of the 2,000 points were independently and randomly generated from a standard normal distribution. We then ran the 2-coordinate descent method (in the one-sided setting of course) with a full minimization step (which in this case is equivalent to the partial minimization step) and compared it to two other alternatives:

- **2R**: the same as the 2-coordinate descent method, but with an index selection strategy which is based on the optimality measure $R$, that is, $j_k \in \operatorname{argmin}_{j=1,\ldots,m} \nabla_j q(\boldsymbol{\lambda}^k)$ and $i_k \in \operatorname{argmax}_{i:\lambda_i^k>0} \nabla_i q(\boldsymbol{\lambda}^k)$.
- **GRAD**: A gradient projection method defined by $\boldsymbol{\lambda}^{k+1} = P_{\Delta_m}(\boldsymbol{\lambda}^k - \frac{1}{L}\nabla q(\boldsymbol{\lambda}^k))$. Here $L$ is the Lipschitz constant of the objective function and is given by $L = 2\lambda_{\max}(\mathbf{A}^T\mathbf{A})$, where $\lambda_{\max}(\mathbf{A}^T\mathbf{A})$ denotes the maximum eigenvalue of the matrix $\mathbf{A}^T\mathbf{A}$.

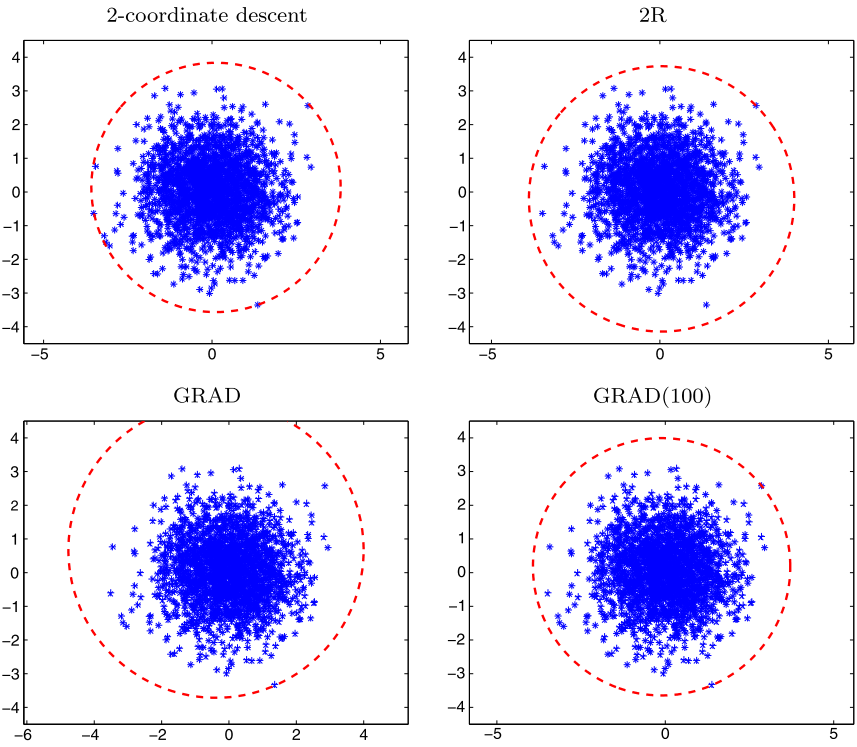


**Fig. 2** The resulting Chebyshev center and circle for different solvers

All the algorithms were initialized with the vector $\boldsymbol{\lambda}^0 = \mathbf{e}_1$ and ran for only 10 iterations. In addition, we present the result of 100 iterations of the gradient projection method; this experiment was denoted by GRAD(100). The resulting circles can be seen in Fig. 2.

We also ran CVX [45] with the SeDuMi solver [46] and found that the optimal radius is 3.6848 and the solution is extremely sparse—it has only three nonzero components in the optimal solution, which correspond to the three points on the boundary of the resulting circle. As can be clearly seen in the four images, the best result was obtained by the 2-coordinate descent method (top left image) with a radius of 3.7001. It can be seen in this image that there are three points on the border of the circle. The result of 2R is clearly worse (radius 3.9382), and the resulting circle is obviously not the minimal one. The gradient projection method GRAD produced a circle which is very far from the optimal one (radius 4.3819), and even when 100 iterations were employed, the result was not satisfactory.

### 7.2 A Random Example

Consider the quadratic minimization problem

$$\min\left\{\frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{b}^T\mathbf{x} : \mathbf{x} \in \Delta_{100}\right\},$$

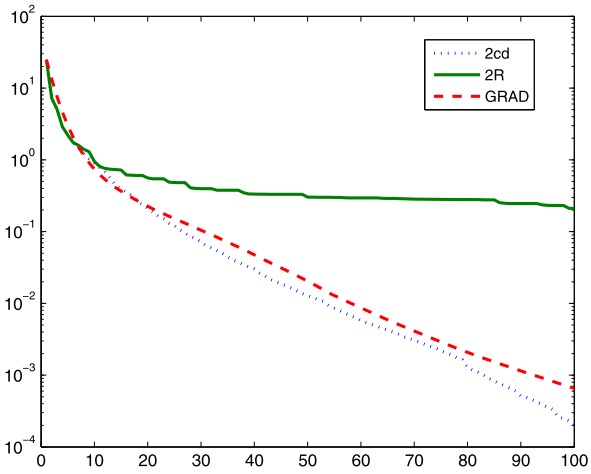


**Fig. 3** Objective function values on log scale of the three solvers

**Table 1** Results for the three methods 2cd, 2R, and GRAD

| Iter. | 2cd | 2R | GRAD | 2cd<2R | 2cd<GRAD |
|---|---|---|---|---|---|
| 10 | 0.9805 | 1.1236 | 1.1692 | 82 | 73 |
| 20 | 0.2761 | 0.5046 | 0.3589 | 98 | 73 |
| 100 | 0.0015 | 0.2036 | 0.0013 | 100 | 49 |

where $\mathbf{Q} = \mathbf{A}\mathbf{A}^T \in \mathbb{R}^{100\times 100}$ with the components of $\mathbf{A} \in \mathbb{R}^{100\times 50}$ being independently and randomly generated from a standard normal distribution. The components of the linear coefficients vector $\mathbf{b}$ were also generated from a standard normal distribution. We employed the three algorithms (2-coordinate descent, 2R, and GRAD) on the resulting problem and computed an exact solution via SeDuMi. The function values of the 100 iterates of the three methods are plotted in Fig. 3.

Clearly, the 2-coordinate descent (2cd) method significantly outperforms 2R, that is, the coordinate descent method with the SVM-type-like index selection strategy, and it provides more accurate solutions than the gradient projection method GRAD. We also ran 100 realizations of the same process, and the results are given in Table 1.

The mean values over 100 realizations of the objective function values of the three methods after 10, 20, and 100 iterations are given in columns 2, 3, and 4. After 10 or 20 iterations the 2cd method is better on average than the other two methods and is slightly inferior to GRAD after 100 iterations. The columns termed `2cd<2R` and `2cd<GRAD` contain the number of runs in which the objective function value obtained by the 2-coordinate descent method was better (i.e., lower) than the objective function value obtained by 2R and GRAD, respectively. Overall, 2cd seems to be a better method than 2R, and its advantage grows when the number of iterations is larger. On the other hand, the last column illustrates that, at least in this set of runs, `2cd` is better than `GRAD` after a small number of iterations (10 or 20), but is not much better after 100 iterations.

### 7.3 Experiments on SVM Training Problems

A well-known problem in classification is the problem of training a support vector machine (SVM) in which one seeks to separate two sets of vectors (or their transformations) by a hyperplane. The dual problem associated with the problem of training an SVM is given by the following convex quadratic programming problem (see [18, 19, 21] for more details):

$$\max\left\{\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j):\sum_{i=1}^{n}\alpha_i y_i=0, 0\le\alpha_i\le C\right\}, \tag{41}$$

where the vectors $\mathbf{x}_1,\ldots,\mathbf{x}_n\in\mathbb{R}^d$ and their corresponding classes $y_1,\ldots,y_n\in\{-1,1\}$ are the given "training data" and $k(\cdot,\cdot)$ is called the kernel function; it is assumed that the matrix $(k(\mathbf{x}_i,\mathbf{x}_j))_{i,j}$ is positive semidefinite. Obviously, problem (41) is of the form of model (1), and indeed many of the methods developed to solve it apply to this model, such as those in [19–24, 29, 40, 44]. In this section we report the results of experiments conducted on some SVM training test problems. The problem that we actually solve is the dual-SVM problem (41). Before applying the algorithm, we transformed it into a problem over a double-sided simplex set by the linear change of variables described in Sect. 2.1. Since the 2-coordinate descent method in the double-sided setting cannot handle large-scale problems, we used an implementable variant of the method, which we call 2cd-hybrid. The index selection strategy in the hybrid method is defined as follows: the index $j$ is predetermined to be chosen as in the maximum violation criteria (9), while the index $i$ is chosen to maximize the same term as in $S_2$. Explicitly, this can be written as:

$$j_k\in\operatorname*{argmin}_{j:x_j<u_j}\nabla_j f\left(\mathbf{x}^k\right), \tag{42}$$

$$i_k\in\operatorname*{argmax}_{i=1,\ldots,n}\left\{\sqrt{\bar{L}_{i,j_k}}\min\left\{\frac{1}{\bar{L}_{i,j_k}}\left[\nabla_i f\left(\mathbf{x}^k\right)-\nabla_{j_k} f\left(\mathbf{x}^k\right)\right], x_i^k-l_i, u_{j_k}-x_{j_k}^k\right\}\right\}. \tag{43}$$

Note that the index selection strategy coincides with the one-sided index selection strategy when the feasible set is indeed a one-sided simplex set. All the data sets have been taken from the LIBSVM database [47]. The problems that were tested along with their dimension are described below:

- a1a ($n=1605$);
- a4a ($n=4781$);
- mushrooms ($n=8124$);
- w5a ($n=9888$);
- w7a ($n=4692$);

To evaluate the performance of the method cd-hybrid, we compared it to two other 2-coordinate descent type methods that differ only in their index selection strategy: (1) the 2R method described in the Chebyshev center example, which is essentially the same as the SVM$^{\text{light}}$ method with $q=2$, although our implementation is probably not as efficient as the implementation in LIBSVM [48], and (2) the method we call

**Table 2** Results for the three methods 2cd-hybrid, 2R and SOI on 5 data sets from LIBSVM

| Problem | Method | $f_{10}$ | $f_{100}$ | $f_{1000}$ | $f_{5000}$ | $f_{10000}$ |
|---|---|---|---|---|---|---|
| a1a | 2cd-hybrid | **−17.17** | **−230.20** | **−664.67** | **−683.41** | **−683.41** |
| | 2R | −11.02 | −160.79 | −658.23 | **−683.41** | **−683.41** |
| | SOI | −17.06 | −219.08 | −663.26 | **−683.41** | **−683.41** |
| a4a | 2cd-hybrid | **−19.17** | **−272.40** | **−1791.65** | **−2358.51** | **−2358.75** |
| | 2R | −16.85 | −185.64 | −1612.24 | −2358.34 | **−2358.75** |
| | SOI | −21.42 | −243.74 | −1785.58 | −2358.48 | **−2358.75** |
| mushrooms | 2cd-hybrid | **−10.01** | **−99.52** | **−675.38** | **−1035.54** | **−1071.64** |
| | 2R | −10.00 | −99.47 | −674.19 | −1035.31 | −1071.50 |
| | SOI | **−10.01** | **−99.52** | **−675.38** | **−1035.54** | **−1071.64** |
| w5a | 2cd-hybrid | **−91.00** | **−460.03** | **−729.68** | −785.10 | −789.37 |
| | 2R | −20.05 | −247.14 | −725.04 | **−785.18** | **−789.39** |
| | SOI | **−91.00** | −459.53 | −729.46 | −785.10 | −789.37 |
| w7a | 2cd-hybrid | **−45.70** | −457.49 | −1817.99 | −2083.18 | −2119.72 |
| | 2R | −21.01 | −243.97 | −1748.78 | −2082.11 | **−2120.07** |
| | SOI | −42.99 | **−492.99** | **−1820.47** | **−2083.51** | −2119.83 |

*SOI* which is the method from [29] where the indices are chosen using additional second order information. The specific index selection strategy is the one called WSS2 in [29].

All the methods were implemented in MATLAB. We used in (41) the Gaussian kernel $k(\mathbf{v}, \mathbf{w}) = e^{-\gamma\|\mathbf{v}-\mathbf{w}\|^2}$, where $\gamma$ was fixed to be 1, and the penalty parameter $C$ was set to 5—this is the same setting that was used in [27]. The three methods are compared in Table 2, where the function values after 10, 100, 1,000, 5,000, and 10,000, denoted by $f_{10}$, $f_{100}$, $f_{1000}$, $f_{5000}$, $f_{10000}$, are given. Note that, from a computational point of view, the three methods are very similar, since they all require access to two columns of the matrix $\mathbf{Q}$ at each iteration, and in addition several operations whose complexity is linear in the dimension. The best results for each of the runs are emphasized in boldface.

Evidently, in the data sets a1a and a4a the 2cd-hybrid algorithm outperforms the other two methods, and SOI seems to give better results, at least until 1,000 iterations, than 2R. For the data set mushrooms, the methods 2cd-hybrid and SOI, give the same results which are better than those obtained by 2R. For w5a, 2cd-hybrid gives the best results at the beginning (at least until the 1,000th iteration), but at iterations 5,000 and 10,000 2R seems be the best option. For the data set w7a, the method SOI gives the best results in iterations 100, 1,000, 5,000, while 2cd-hybrid gives the best results after 10 iterations and 2R is slightly better than the other two approaches in iteration 10,000. To summarize, the hybrid 2-coordinate descent method produced improved the results in most of the scenarios that were tested.

## 8 Conclusions

In this paper, we considered the problem of minimizing a continuously differentiable function with a Lipschitz continuous gradient over a single linear equality constraint and bound constraints. Based on new optimality measures, we were able to derive new block descent methods that perform at each iteration an optimization procedure on two chosen decision variables. In the convex case, the main result is a nonasymptotic sublinear rate of convergence of the function values. There are still several open and interesting research questions that can be investigated. First, can the analysis be generalized to the interesting and general case, where the constraint set consists of an arbitrary number of equality constraints? This will require a generalization of both the optimality measures, the index selection strategies and the convergence analysis. Second, the rate of convergence analysis is restricted to the one-sided setting, and a generalization to the two-sided setting does not seem to be straightforward; therefore, the question that arises is: Does there exist another proof technique that will enable us to analyze this important setting as well? A final important open question is: Can the dependency of the efficiency estimate in the dimension of the problem (Theorem 6.1) be removed?

**Acknowledgements** I would like to thank the three anonymous reviewers for their useful comments and additional references which helped to improve the presentation of the paper. This work was partially supported by ISF grant #25312 and by BSF grant 2008100.

## References

1. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation. Prentice-Hall, Englewood Cliffs (1989)
2. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. Academic Press, New York (1970)
3. Auslender, A.: Méthodes numériques pour la décomposition et la minimisation de fonctions non différentiables. Numer. Math. **18**, 213–223 (1971/72)
4. Auslender, A.: Optimisation. Méthodes Numériques, Maîtrise de Mathématiques et Applications Fondamentales. Masson, Paris (1976)
5. Auslender, A., Martinet, B.: Méthodes de décomposition pour la minimisation d'une fonctionnelle sur un espace produit. C. R. Acad. Sci. Paris Sér. A-B **274**, A632–A635 (1972)
6. Bertsekas, D.P.: Nonlinear Programming, 2nd edn. Athena Scientific, Belmont (1999)
7. Cassioli, A., Lorenzo, D.D., Sciandrone, M.: On the convergence of inexact block coordinate descent methods for constrained optimization. Eur. J. Oper. Res. **231**(2), 274–281 (2013)
8. Cassioli, A., Sciandrone, M.: A convergent decomposition method for box constrained optimization problems. Optim. Lett. **3**(3), 397–409 (2009)
9. Grippo, L., Sciandrone, M.: Globally convergent block-coordinate techniques for unconstrained optimization. Optim. Methods Softw. **10**, 587–637 (1999)
10. Luo, T., Tseng, P.: Error bounds and convergence analysis of feasible descent methods: a general approach. Ann. Oper. Reas. **46**, 157–178 (1993)
11. Powell, M.J.D.: On search directions for minimization algorithms. Math. Program. **4**, 193–201 (1973)
12. Luo, T., Tseng, P.: On the convergence of the coordinate descent method for convex differentiable minimization. J. Optim. Theory Appl. (1992)
13. Polak, E., Sargent, R.W.H., Sebastian, D.J.: On the convergence of sequential minimization algorithms. J. Optim. Theory Appl. **14**, 439–442 (1974)
14. Sargent, R.W.H., Sebastian, D.J.: On the convergence of sequential minimization algorithms. J. Optim. Theory Appl. **12**, 567–575 (1973)

15. Tseng, P., Yun, S.: A coordinate gradient descent method for nonsmooth separable minimization. Math. Program. **117**, 387–423 (2009)
16. Nesterov, Y.: Efficiency of coordinate descent methods on huge-scale optimization problems (2010). CORE Discussion paper 2010/2
17. Beck, A., Tetruashvili, L.: On the convergence of block coordinate descent type methods. SIAM J. Optim. **23**(2), 2037–2060 (2013)
18. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. Data Min. Knowl. Discov. **2**, 121–167 (1998)
19. Platt, J.C.: Sequential minimal optimization: a fast algorithm for training support vector machines. In: Scholkopf, B., Burges, C.J.C., Smola, A.J. (eds.) Advances in Kernel Methods—Support Vector Learning, pp. 185–208. MIT Press, Cambridge (1999)
20. Keerthi, S., Gilbert, E.: Convergence of a generalized SMO algorithm for SVM. Mach. Learn. **46**, 351–360 (2002)
21. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: Improvements to Platt's SMO algorithm for SVM classifier design. Neural Comput. **13**(3), 637–649 (2001)
22. Joachims, T.: Making large-scale SVM learning practical. In: Scholkopf, B., Burges, C.J.C., Smola, A.J. (eds.) Advances in Kernel Methods—Support Vector Learning, B, pp. 169–184. MIT Press, Cambridge (1999)
23. Lin, C.J.: On the convergence of the decomposition method for support vector machines. IEEE Trans. Neural Netw. **12**, 1288–1298 (2001)
24. Lin, C.J.: Asymptotic convergence of an SMO algorithm without any assumptions. IEEE Trans. Neural Netw. **13**, 248–250 (2002)
25. Palagi, L., Sciandrone, M.: On the convergence of a modified version of $\text{SVM}^{\text{light}}$ algorithm. Optim. Methods Softw. **20**(2–3), 317–334 (2005)
26. Lucidi, S., Palagi, L., Risi, A., Sciandrone, M.: A convergent hybrid decomposition algorithm model for SVM training. IEEE Trans. Neural Netw. **20**(6), 1055–1060 (2009)
27. Lin, C.J., Lucidi, S., Palagi, L., Risi, A., Sciandrone, M.: Decomposition algorithm model for singly linearly-constrained problems subject to lower and upper bounds. J. Optim. Theory Appl. **141**(1), 107–126 (2009)
28. Lucidi, S., Palagi, L., Risi, A., Sciandrone, M.: A convergent decomposition algorithm for support vector machines. Comput. Optim. Appl. **38**(2), 217–234 (2007)
29. Chen, P.H., Fan, R.E., Lin, C.J.: Working set selection using second order information for training support vector machines. J. Mach. Learn. Res. **6**, 1889–1918 (2005)
30. Glasmachers, T., Igel, C.: Maximum-gain working set selection for SVMs. J. Mach. Learn. Res. **7**, 1437–1466 (2006)
31. Chang, C.C., Hsu, C.W., Lin, C.J.: The analysis of decomposition methods for support vector machines. IEEE Trans. Neural Netw. **11**, 1003–1008 (2000)
32. Dai, Y.H., Fletcher, R.: New algorithms for singly linearly constrained quadratic programs subject to lower and upper bounds. Math. Program., Ser. A **106**(3), 403–421 (2006)
33. Serafini, T., Zanghirati, G., Zanni, L.: Gradient projection methods for quadratic programs and applications in training support vector machines. Optim. Methods Softw. **20**, 353–378 (2005)
34. Tseng, P., Yun, S.: A coordinate gradient descent method for linearly constrained smooth optimization and support vector machines training. Comput. Optim. Appl. **47**(2), 179–206 (2010)
35. Liuzzi, G., Palagi, L., Piacentini, M.: On the convergence of a Jacobi-type algorithm for singly linearly-constrained problems subject to simple bounds. Optim. Lett. **5**(2), 347–362 (2011)
36. Markowitz, H.: Portfolio selection. J. Finance **7**, 77–91 (1952)
37. Bomze, I.M.: Evolution towards the maximum clique. J. Glob. Optim. **10**(2), 143–164 (1997)
38. Xu, S., Freund, R.M., Sun, J.: Solution methodologies for the smallest enclosing circle problem. Comput. Optim. Appl. **25**(1–3), 283–292 (2003). A tribute to Elijah (Lucien) Polak
39. Lin, C.J.: A formal analysis of stopping criteria of decomposition methods for support vector machines. IEEE Trans. Neural Netw. **13**(5), 1045–1052 (2002)
40. Hush, D., Scovel, C.: Polynomial-time decomposition algorithms for support vector machines. Mach. Learn. **51**(1), 51–71 (2003)
41. Beck, A., Teboulle, M.: Gradient-based algorithms with applications to signal recovery problems. In: Eldar, Y., Palomar, D. (eds.) Convex Optimization in Signal Processing and Communications. Cambridge University Press, Cambridge (2010)
42. Nesterov, Y.: Introductory Lectures on Convex Optimization. Kluwer, Boston (2004)
43. Polyak, B.T.: Introduction to Optimization. Translations Series in Mathematics and Engineering. Optimization Software, New York (1987)

44. Chen, P.H., Fan, R.E., Lin, C.J.: A study on SMO-type decomposition methods for support vector machines. IEEE Trans. Neural Netw. **17**(4), 893–908 (2006)
45. Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 1.21. http://cvxr.com/cvx (2011)
46. Sturm, F.J.: Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. Optim. Methods Softw. **11–12**, 625–653 (1999)
47. Chang, C.C., Lin, C.J.: LIBSVM data: Classification, regression, and multi-label. http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/
48. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 1–27 (2011). Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm