ISSN: 1392-6292/ eISSN: 1648-3510

VILNIUS TECH
Vilnius Gediminas Technical University

MATHEMATICAL MODELLING and ANALYSIS

2025
Volume 30
Issue 3
Pages 535–552
https://doi.org/10.3846/mma.2025.23064

A partially derivative-free cyclic block coordinate descent method for nonseparable composite optimization

Vitaliano Amaral,

Departamento de Matemática Teresina, Universidade Federal do Piauí, PI, Brazil

Article History:
- received January 8, 2025
- revised May 7, 2025
- accepted June 11, 2025

Abstract. In this paper, we address a composite optimization problem in which the objective function consists of two terms: the first presents a function with a gradient that satisfies a Lipschitz–Hölder composition, while the second one is a convex function. Under general settings, we propose and analyze a new coordinate descent method that can operate without the use of derivatives. The algorithm is an adaptation of the coordinate proximal gradient method, specifically designed to consider the composite form of the objective function. We perform a complete worst-case complexity analysis, assuming that the coordinates (or blocks of coordinates) are selected in a cyclic manner. In addition, we present academic numerical examples that illustrate the efficiency of our algorithm in practical problems.

Keywords: coordinate descent methods; worst-case evaluation complexity; composite minimization; nonseparable objective function.

AMS Subject Classification: 90C30; 65K05; 49M37; 90C60; 68Q25.

✉ Corresponding author. E-mail: vitalianoamaral@ufpi.edu.br

1 Introduction

In this paper, we consider the unconstrained optimization problem

$$\text{Minimize } F(x) := f(x) + h(x) \text{ subject to } x \in \mathbb{R}^n, \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a function of the form $f := g_1 + g_2$, where g_1 has a continuous Lipschitz gradient, g_2 has a continuous Hölder gradient, and $h : \mathbb{R}^n \to \mathbb{R}$ is a convex function. When h is the indicator function of a convex set C then Problem (1.1) is equivalent to solving the problem of minimizing $F(x) = f(x)$ constrained to the convex set C.

Many optimization problems in real-world applications are often large-scale and high-dimensional, which makes them challenging to solve, especially when there are derivatives that are difficult to calculate. Popular methods for addressing these problems include sparse models, as demonstrated in recent publi-

cations on machine learning, statistics, and signal processing. Due to the structure of many such problems, derivative-free block coordinate descent (BCD) methods are particularly suitable, as they utilize only a few variables in each iteration, reducing the computational cost associated with function value and gradient calculations.

The central idea of coordinate descent methods is to decompose a large optimization problem into a sequence of smaller problems, thereby reducing the computational effort required during the method's execution. This class of methods was among the first to appear in the literature on variable decomposition, with its roots in pioneering algorithms such as the one described by Gauss and Seidel for the minimization of quadratic functions. Recent studies have shown that these methods are effective in handling high-dimensional problems with moderate accuracy, sparking growing interest in various applications [1, 2, 4, 5, 8, 9, 11]. In this context, extensions of BCD methods become particularly relevant, considering the emergence of large-scale problems in different areas.

In [11], the convergence of BCD methods was analyzed in the context of convex function minimization, with a focus on machine learning applications. The study highlights the relevance of random selection of variable blocks at each iteration, exploring optimized parallel implementations that demonstrate the efficiency of BCD methods in this scenario. Furthermore, convergence properties similar to those of the deterministic cyclic BCD described in [2] were established.

In [1], a version of BCD methods with higher-order regularization was proposed to minimize smooth, possibly non-convex functions under box constraints. The study presented complexity results, showing that the BCD method with $p+1$ order regularization requires at most $O(\epsilon^{-(p+1)})$ outer iterations to reach an ϵ-stationary point, where the 2-norm of the gradient of f is less than ϵ. In [9], a convergence analysis of the Randomized Block Coordinate Descent method for smooth block Hölder functions was presented, covering non-convex, convex and strongly convex cases. It was shown that, for non-convex functions, the expected norm of the gradient reduces to $\mathcal{O}\left(k^{\frac{\beta}{1+\beta}}\right)$, where k is the number of iterations and β-the Hölder exponent. In the convex case the reduction fo is of order $\mathcal{O}\left(k^{-\beta}\right)$, and in the strongly convex case, the reduction improves to $\mathcal{O}\left(k^{-\frac{2\beta}{1-\beta}}\right)$, when $\beta > 1$, and reaches a linear rate for $\beta = 1$.

In the context of methods with low computational cost, the search for derivative-free alternatives has also advanced significantly. For instance, in [6,7] and the references therein, Grapiglia proposed in [7] a quadratic regularization method to minimize a function $f : \mathbb{R}^n \to \mathbb{R}$. In this method, finite difference approximations of gradients were used, demonstrating that for functions bounded below with Lipschitz gradients, the method requires at most $\mathcal{O}(\epsilon^{-2})$ iterations to generate an approximate stationary point with accuracy ϵ.

In this work, we propose a BCD method for the optimization problem (1.1), assuming that the gradient of f satisfies a Hölder-Lipschitz condition and that h is convex. We show that the proposed method requires at most $\mathcal{O}\left(\epsilon^{-\frac{\beta+1}{\beta}}\right)$

iterations to find an approximate solution with tolerance ϵ, where β is the exponent of the assumed Hölder condition. Our complexity bound is consistent with that established by Martínez in [10] for the first-order version of his p-order method. Furthermore, when $\beta = 1$ (Lipschitz condition), we obtain a complexity of $\mathcal{O}\left(\epsilon^{-2}\right)$, in accordance with the results presented in [1, 7].

We chose a cyclic structure in our method, as it naturally guarantees that all blocks are updated throughout the execution - a condition often required in theoretical analyses of convergence and complexity, as discussed in [1]. This structure also facilitates the extension of the convergence analysis to non-convex scenarios, which is in line with the goals of this work. Furthermore, the cyclic scheme simplifies the analysis under Hölder-type smoothness assumptions.

The remainder of this paper is organized as follows. In Section 2, we introduce the notations and the main preliminary results that support our contributions. In Section 3, we present a detailed description of the BCD method, accompanied by an analysis of its effectiveness. Section 4 is devoted to the analysis of the worst-case iteration complexity of the method. In Section 5, numerical examples are presented. Finally, in Section 6, we discuss the conclusions of the study.

2 Notation and auxiliary results

In this section, we present some definitions and results that are essential for understanding the remainder of this work.

Throughout this text, we use the following symbols: $\langle \cdot, \cdot \rangle$ represents the usual inner product, $\| \cdot \|$ denotes the Euclidean norm and $\| \cdot \|_\infty$ represents the sup-norm.

The dimension of the i-th block of variables is denoted by n_i, where $i = 1, \ldots, q$, and satisfies $n_1 + n_2 + \ldots + n_q = n$. The matrices U_i, for $i = 1, \ldots, q$, are chosen such that $\left[U_1,\ U_2,\ \ldots,\ U_q\right] = I_n$, where I_n is the $n \times n$ identity matrix.

Each matrix $U_i \in \mathbb{R}^{n \times n_i}$ is used to extract the vector $v_i = U_i^T v \in \mathbb{R}^{n_i}$, which consists of the components of $v \in \mathbb{R}^n$ corresponding to the i-th block.

With this setup, the *partial gradient* of a function $g : \mathbb{R}^n \to \mathbb{R}$ can be written as $\nabla_{(i)} g(x) = U_i^T \nabla g(x)$, derived from the total gradient $\nabla g(x)$. Similarly, the *partial Hessian* of g at x is given by $\nabla^2_{(i)} g(x) = U_i^T \nabla^2 g(x) U_i$, which is a matrix of dimension $n_i \times n_i$.

3 BCD with quadratic regularization

The cyclic block coordinate descent method is presented in Algorithm 1.

For the development of the method proposed in this work we consider a function φ_f that satisfies

$$\varphi_f : \mathbb{R}^n \times [0, 1] \to \mathbb{R}^n \text{ where } \lim_{\lambda \to 0} \varphi_f(x, \lambda) = \nabla f(x). \tag{3.1}$$

The function in (3.1) can be defined as $\varphi_f = \nabla f$. However, this definition is not recommended when the computation of the gradient of f is computationally

prohibitive. In such cases, it is preferable to define φ_f as an approximation of the gradient ∇f, which can be obtained at a lower computational cost.

Below we present three options for the definition of φ_f.

Remark 1. If f is differentiable, then φ_f can be defined in one of the following ways:

$$\varphi_f(x,\lambda) = \left[\frac{f(x+\lambda e_1)-f(x)}{\lambda},\cdots,\frac{f(x+\lambda e_n)-f(x)}{\lambda}\right] \text{ and } \varphi_f(x,0)=\nabla f(x),$$

$$\varphi_f(x,\lambda) = \left[\frac{f(x)-f(x-\lambda e_1)}{\lambda},\cdots,\frac{f(x)-f(x-\lambda e_n)}{\lambda}\right] \text{ and } \varphi_f(x,0)=\nabla f(x),$$

$$\varphi_f(x,\lambda) = \left[\frac{f(x+\lambda e_1)-f(x-\lambda e_1)}{2\lambda},\cdots,\frac{f(x+\lambda e_n)-f(x-\lambda e_n)}{2\lambda}\right] \text{ and } \varphi_f(x,0)=\nabla f(),$$

where e_i, $i=1,\ldots,n$ is the canonical vector of $\mathbb{R}^n$. In practice, the central difference formula is the most accurate.

Below we present a version of a partially derivative-free method, which can be made completely derivative-free, to solve Problem (1.1).

Algorithm 1. **- Derivative-free Block Cyclic Coordinate Descent - BCDC-Dfree**

Let $x^0 \in \mathbb{R}^n$, for each $i=1,\ldots,q$ a symmetric positive semidefinite matrix $B_{(i)}(x^0) \in \mathbb{R}^{n_i \times n_i}$, $\alpha \in (0,1)$, $\epsilon \in (0,1)$, $\sigma_0 \geq 1$ and $F_{\text{target}} \in \mathbb{R}$.
Initialize $k \leftarrow 0$.

Step 1: For $\lambda_k \in \left[0, \epsilon/\sigma_k\sqrt{n}\right]$ consider $\varphi_f(x^k,\lambda_k)$.

Step 2: Write $x^{k,0}=x^k$ and for each $i=1,\ldots,q$ compute

$$x^{k,i} = x^{k,i-1} + U_i s^k_{(i)},$$

where $s^k_{(i)} \in \mathbb{R}^{n_i}$ is a solution to the following problem

$$\min_{s\in\mathbb{R}^{n_i}} \langle U_i^T \varphi_f(x^{k,i-1},\lambda_k), s\rangle + \frac{1}{2}\langle B_{(i)}(x^{k,i-1})s, s\rangle + h(x^k+U_i,s) - h(x^{k,i-1}) + \frac{\sigma_k}{2}\|s\|^2. \tag{3.2}$$

Step 3. If $\|\sum_{i=1}^{q} U_i s^k_{(i)}\|_\infty < \dfrac{\epsilon}{\sigma_k}$ or $F(x^{k,q}) \leq F_{\text{target}}$, stop declaring $x^{k,q}$ an acceptable solution. Otherwise, go to Step 4.

Step 4: If

$$F(x^{k,q}) \leq F(x^k) - \frac{\alpha}{\sigma_k}\epsilon^2 \tag{3.3}$$

holds, take $k \leftarrow k+1$, define $x^{k+1}=x^{k,q}$ and $\sigma_{k+1}=\sigma_k$, choose a symmetric positive semidefinite matrix $B_{(i)}(x^{k+1}) \in \mathbb{R}^{n_i\times n_i}$ and go to Step 1. Otherwise, define $\sigma_k \leftarrow 2\sigma_k$ and go to Step 1.

The parameter α controls the level of reduction of F in (3.3), where values close to 1 result in a more aggressive decrease e F, which may cause the penalty parameter σ_k to grow unnecessarily. Therefore, it is crucial to adjust α carefully. The descent test in (3.3) shows that small values of σ_k tend to produce larger

steps, which can accelerate the convergence of the method. The justification for the stopping criterion adopted in Step 3 is provided in Remark 3.

It is important to note that the matrix $B_{(i)}(x^{k,i-1})$ used in Algorithm 1 does not need to be an approximation of the Hessian $\nabla^2_{(i)} f(x^{k,i-1})$, as the expression for g_i^k might suggest. It can even be defined as the null matrix, where only the first-order information is used, which is useful when the Hessian $\nabla^2_{(i)} f(x^{k,i-1})$ is expensive to compute. However, second-order information, when available, can improve the efficiency of the step. The flexibility in choosing $B_{(i)}(x^{k,i-1})$ allows using the Hessian, a cheap approximation (e.g., quasi-Newton), or the null matrix, as long as it is uniformly bounded (Assumption 2).

To obtain the good definition of Step 2, for each $i \in \{1, 2, \cdots, q\}$ we consider the function $g_i^k : \mathbb{R}^{n_i} \to \mathbb{R}$ where

$$g_i^k(s)=\langle U_i^T \varphi_f(x^{k,i-1}, \lambda_k)+\frac{1}{2}B_{(i)}(x^{k,i-1})s, s\rangle+h(x^k+U_i s)-h(x^{k,i-1})+\frac{\sigma_k}{2}\|s\|^2$$

with $B_j^k(x^{k,i-1})$ symmetric matrix. Since h is convex, then there exists a linear function $\overline{h}(x) = \langle a, x\rangle + b$, $a, x \in \mathbb{R}^n$, $b \in \mathbb{R}$ such that $h(x) \geq \overline{h}(x)$ for all $x \in \mathbb{R}^n$. For simplicity we write $H(s) = \langle U_i^T \varphi_f(x^{k,i-1}, \lambda_k), s\rangle + \frac{\sigma_k}{2}\|s\|^2$. From which it follows that

$$\begin{aligned}
&H(s) + \frac{\sigma_k \delta_{ki}^{\min}}{2}\|s\|^2 + \langle a, x^k + U_i s\rangle + b - h(x^{k,i-1}) \\
&\leq H(s) + \frac{1}{2}\langle B_{(i)}(x^{k,i-1})s, s\rangle + \langle a, x^k + U_i s\rangle + b - h(x^{k,i-1}) \\
&\leq H(s) + \frac{1}{2}\langle B_{(i)}(x^{k,i-1})s, s\rangle + \overline{h}(x^k + U_i s) - h(x^{k,i-1}) \\
&\leq H(s) + \frac{1}{2}\langle B_{(i)}(x^{k,i-1})s, s\rangle + h(x^k + U_i s) - h(x^{k,i-1})
\end{aligned}$$

holds for all s in $\mathbb{R}^{n_i}$ and fixed $x^{k,i-1}$, where $\delta_{ki}^{\min}$ is the smallest eigenvalue of $B_{(i)}(x^{k,i-1})$. Therefore, we can write

$$+\infty= \lim_{\|s\|\to\infty}\left(H(s)+\langle a, x^k+U_i s\rangle + b-h(x^{k,i-1})+\frac{\sigma_k\delta_{ki}}{2}\|s\|^2\right) \leq \lim_{\|s\|\to\infty} g_i^k(s),$$

which implies that $g_i^k(\cdot)$ is coercive, thus ensuring that each Subproblem in (3.2) has a solution. Therefore ensuring the well-definedness of Step 2.

In the following section, we analyze the satisfiability of (3.3) in Step 4. To this end, we consider the following assumption.

Assumption 1 *There are $L, M \in (0, +\infty)$ and $\beta \in (0, 1]$ such that*

$$f(y) \leq f(x)+\langle \nabla f(x), y-x\rangle+\frac{L}{2}\|y-x\|^2+\frac{M}{\beta+1}\|y-x\|^{\beta+1} \ \forall\, x, y \in \mathbb{R}^n \quad (3.4)$$

and

$$\|\nabla f(x) - \varphi_f(x, \lambda)\| \leq \frac{\sqrt{n}L}{2}\lambda + \frac{\sqrt{n}M}{\beta+1}\lambda^{\beta}. \quad (3.5)$$

Remark 2. If $f := g_1 + g_2$, where g_1 has a continuous L-Lipschitz gradient and g_2 has a continuous M-Hölder gradient with exponent β, we have that

$$g_1(y) \leq g_1(x) + \langle \nabla g_1(x), y - x \rangle + \frac{L}{2}\|y - x\|^2 \ \forall\, x, y \in \mathbb{R}^n,$$
$$g_2(y) \leq g_2(x) + \langle \nabla g_2(x), y - x \rangle + \frac{M}{\beta + 1}\|y - x\|^{\beta+1} \ \forall\, x, y \in \mathbb{R}^n.$$

See e.g. [12, Lemma 1], which directly implies (3.4).

In the following Lemma, we will show that if f satisfies the same conditions given in the Remark 2, then the condition (3.5) holds.

Lemma 1. *The conditions* (3.4) *hold if* $f := g_1 + g_2$, *where* g_1 *has a continuous* L*-Lipschitz gradient and* g_2 *has a continuous* M*-Hölder gradient with exponent* β. *Additionally,* (3.5) *is satisfied by considering* $\varphi_g(x, \lambda)$ *as any of the definitions given in Remark 1.*

Proof. Considering g_1 with a continuous L-Lipschitz gradient and g_2 with a continuous M-Holder gradient with exponent β, we can conclude that

$$g_1(y) \leq g_1(x) + \langle \nabla g_1(x), y - x \rangle + \frac{L}{2}\|y - x\|^2 \ \forall\, x, y \in \mathbb{R}^n,$$
$$g_2(y) \leq g_2(x) + \langle \nabla g_2(x), y - x \rangle + \frac{M}{\beta + 1}\|y - x\|^{\beta+1} \ \forall\, x, y \in \mathbb{R}^n.$$

By summing the two previous inequalities, we obtain (3.4). Furthermore, we obtain

$$\|g_1(y) - g_1(x) - \nabla g_1(x)^T(y - x)\| \leq \frac{L}{2}\|y - x\|^2,$$
$$\|g_2(y) - g_2(x) - \nabla g_2(x)^T(y - x)\| \leq \frac{M}{1 + \beta}\|y - x\|^{\beta+1}.$$

1. Using $y = x + \lambda e_i$ in the last inequality, we have:

$$\left| \frac{g_1(x + \lambda e_i) - g_1(x)}{\lambda} - \frac{\partial g_1(x)}{\partial x_i} \right| \leq \frac{L}{2}\lambda,$$
$$\left| \frac{g_2(x + \lambda e_i) - g_2(x)}{\lambda} - \frac{\partial g_2(x)}{\partial x_i} \right| \leq \frac{M}{1 + \beta}\lambda^{\beta},$$

this implies that

$$\|\nabla g_1(x) - \varphi_{g_1}(x, \lambda)\| \leq \frac{\sqrt{n}L}{2}\lambda \text{ e } \|\nabla g_2(x) - \varphi_{g_2}(x, \lambda)\| \leq \frac{\sqrt{n}M}{\beta + 1}\lambda^{\beta}.$$

2. Using $y = x - \lambda e_i$ in the last inequality, we have:

$$\left| \frac{g_1(x) - g_1(x - \lambda e_i)}{\lambda} - \frac{\partial g_1(x)}{\partial x_i} \right| \leq \frac{L}{2}\lambda,$$
$$\left| \frac{g_2(x) - g_2(x - \lambda e_i)}{\lambda} - \frac{\partial g_2(x)}{\partial x_i} \right| \leq \frac{M}{1 + \beta}\lambda^{\beta},$$

this implies that

$$\|\nabla g_1(x) - \varphi_{g_1}(x,\lambda)\| \leq \frac{\sqrt{n}L}{2}\lambda \text{ e } \|\nabla g_2(x) - \varphi_{g_2}(x,\lambda)\| \leq \frac{\sqrt{n}M}{\beta+1}\lambda^{\beta}.$$

3. For the central difference, we obtain the following

$$\begin{aligned}\left|\frac{g_1(x+\lambda e_i) - g_1(x+\lambda e_i)}{2\lambda} - \frac{\partial g_1(x)}{\partial x_i}\right| \leq & \frac{1}{2}\left|\frac{g_1(x+\lambda e_i) - g_1(x)}{\lambda} - \frac{\partial g_1(x)}{\partial x_i}\right| \\ + \frac{1}{2}\left|\frac{g_1(x) - g_1(x-\lambda e_i)}{\lambda} - \frac{\partial g_1(x)}{\partial x_i}\right| \leq & \frac{1}{2}\left(\frac{\sqrt{n}L}{2}\lambda + \frac{\sqrt{n}L}{2}\lambda\right) = \frac{\sqrt{n}L}{2}\lambda.\end{aligned}$$

From there, we have that

$$\|\nabla g_1(x) - \varphi_{g_1}(x,\lambda)\| \leq 0.5\sqrt{n}L\,\lambda.$$

In a similar manner, we obtain

$$\|\nabla g_2(x) - \varphi_{g_2}(x,\lambda)\| \leq \frac{\sqrt{n}M}{\beta+1}\lambda^{\beta}.$$

From which it follows that

$$\begin{aligned}\|\nabla g(x) - \varphi_f(x,\lambda)\| &\leq \|\nabla g_1(x) - \varphi_{g_1}(x,\lambda)\| + \|\nabla g_2(x) - \varphi_{g_2}(x,\lambda)\| \\ &\leq \frac{\sqrt{n}L}{2}\lambda + \frac{\sqrt{n}M}{\beta+1}\lambda^{\beta}.\end{aligned}$$

Concluding the proof of the lemma. □

One of the main motivations for this work is the problem of minimizing a nonlinear least squares function, penalized by a norm L_p, with $1 < p < 2$. This problem is associated with a data assimilation context, as defined below:

$$\min F(x) = \frac{1}{2}\|A(x) - b\|_2^2 + \frac{\lambda}{p}\|\Phi(x)\|_p^p, \tag{3.6}$$

where $A : \mathbb{R}^n \to \mathbb{R}^m$ is a differentiable function that is not necessarily linear, $b \in \mathbb{R}^m$, and Φ is a linear operator. For more details and motivation regarding the above problem, see [3].

As we can see, if $0.5\|A(x) - b\|_2^2$ is a convex function, then problem (3.6) is a special case of problem (1.1) with

$$h(x) = 0.5\|A(x) - b\|_2^2, \text{ and } f(x) = \frac{\lambda}{p}\|\Phi(x)\|_p^p.$$

It is shown in [3] that $g(x) = \frac{\lambda}{p}\|\Phi(x)\|_p^p$ has a continuous gradient $(p-1)$-Hölder in x. Another important observation is that in the case where A is linear, we have a special case of the problem (1.1) with

$$g = g_1 + g_2 \text{ where } g_1(x){=}0.5\|A(x) - b\|_2^2, \quad g_2(x){=}\frac{\lambda}{p}\|\Phi(x)\|_p^p, \quad f{=}h{\equiv}0.$$

For, in this case, g_1 has a Lipschitz gradient and g_2 has a Hölder gradient, thus satisfying Assumption 1.

In the following remark (Remark 3), we will provide a choice of parameter λ_k such that the stopping criterion $\bar{x}^{k+1} - x^k = 0$ implies that

$$d\big(0, \partial h(x^k) + \nabla f(x^k)\big) < \epsilon.$$

Remark 3. Since $s^k_{(i)}$ is a solution of the (3.2), then there is $w^k_i \in \partial h(x^{k,q})_{(i)}$ such that

$$U_i^T \varphi_f(x^{k,i-1}, \lambda_k) + B_{(i)}(x^{k,i-1}) s^k_{(i)} + w^k_i + 2\sigma_k s^k_{(i)} = 0,$$

therefore $s^k_{(i)} = 0$, $i = 1, \ldots, q$ implies that

$$U_i^T \varphi_f(x^{k,i-1}, \lambda_k) + w^k_i = 0,\ i = 1, \ldots, q. \tag{3.7}$$

By (3.7) and Assumption 1 we have

$$\begin{aligned} \|\nabla_{(i)} f(x^k) + w^k_i\| &\leq \|U_i^T \varphi_f(x^k, \lambda_k) + w^k_i\| + \frac{\sqrt{n}L}{2}\lambda_k + \frac{\sqrt{n}M}{\beta+1}\lambda_k^\beta \\ &\leq \frac{\sqrt{n}L}{2}\lambda_k + \frac{\sqrt{n}M}{\beta+1}\lambda_k^\beta,\ i = 1, \ldots, q. \end{aligned}$$

If $\lambda_k \leq \min\left\{\dfrac{\epsilon}{qL\sqrt{n}}, \left(\dfrac{(\beta+1)\epsilon}{2qM\sqrt{n}}\right)^{\frac{1}{\beta}}\right\}$, then from the inequality above we have that

$$\|\nabla_{(i)} f(x^k) + w^k_i\| \leq \frac{\epsilon}{2q} + \frac{\epsilon}{2q} \leq \frac{\epsilon}{q},\ i = 1, \ldots, q.$$

This implies that

$$\begin{aligned} d(0, \nabla f(x^k) + \partial h(x^{k,q})) &\leq \sum_{i=1}^{q} d(0, \nabla_{(i)} f(x^k) + \partial h(x^{k,q})_{(i)}) \\ &\leq \sum_{i=1}^{q} \|\nabla_{(i)} f(x^k) + w^k_i\| \leq \sum_{i=1}^{q} \frac{\epsilon}{q} = \epsilon. \end{aligned}$$

By Step 2 we have that $s^k_{(i)} = 0$, $i = 1, \ldots, q$ implies that $x^{k,q} = x^k$. Thus, $\lambda_k \leq \min\left\{\dfrac{\epsilon}{qL\sqrt{n}}, \left(\dfrac{(\beta+1)\epsilon}{2qM\sqrt{n}}\right)^{\frac{1}{\beta}}\right\}$ and $s^k = 0$ implies that

$$d(0, \nabla f(x^k) + \partial h(x^{k,q})) \leq \epsilon.$$

The previous inequality shows that $s^k \approx 0$ implies that x^k approximates an ϵ-stationary point. This suggests that we can adopt $\sigma_k \|s^k\|_\infty < \epsilon$, for some $\epsilon > 0$, as a stopping criterion for the Algorithm 1.

The following Assumption and the next Lemma are useful for demonstrating that the sufficient descent condition in Step 4 of Algorithm 1 is satisfied for all σ_k sufficiently large, as established in Theorem 3.3.

Assumption 2 *There is $\overline{B} \geq 1$ such that $\|B_{(i)}(x^{k,i-1})\| \leq \overline{B}$ for all $i \in \{1,\ldots,q\}$ and k.*

Lemma 2. *Let $x^{k,q}$ be obtained in Step 2 of the Algorithm 1. If Assumptions 1 and 2 hold, and $\|s^k\|_\infty \geq \frac{\epsilon}{\sigma_k}$. Then,*

$$F(x^{k,q}) \leq F(x^k)+\frac{q(Mn^{\frac{1-\beta}{2}}+M)}{\beta+1}\|s^k\|_\infty^{\beta+1}+\left(qL+\frac{q\overline{B}}{2}\right)\|s^k\|_\infty^2-\frac{\sigma_k}{2}\|s^k\|_\infty^2.$$

Proof. By Assumption 1 we have

$$\begin{aligned}
&f(x^{k,i})+h(x^{k,i})\\
&\leq f(x^{k,i-1})+h(x^{k,i})+\langle\nabla_{(i)}f(x^{k,i-1}),s^k_{(i)}\rangle+\frac{L}{2}\|s^k_{(i)}\|^2+\frac{M}{\beta+1}\|s^k_{(i)}\|^{\beta+1}\\
&\leq f(x^{k,i-1})+\langle\varphi_f(x^{k,i-1},\lambda_k),s^k_{(i)}\rangle+\frac{1}{2}\langle B_{(i)}(x^{k,i-1})s^k_{(i)},s^k_{(i)}\rangle+h(x^{k,i})-h(x^{k,i-1})\\
&+\frac{\sigma_k}{2}\|s^k_{(i)}\|^2+h(x^{k,i-1})+\langle\nabla_{(i)}f(x^{k,i-1}),s^k_{(i)}\rangle+\frac{L}{2}\|s^k_{(i)}\|^2+\frac{M}{\beta+1}\|s^k_{(i)}\|^{\beta+1}\\
&-\langle g_{(i)}(x^{k,i-1},\lambda_k),s^k_{(i)}\rangle-\frac{1}{2}\langle B_{(i)}(x^{k,i-1})s^k_{(i)},s^k_{(i)}\rangle-\frac{\sigma_k}{2}\|s^k_{(i)}\|^2. \qquad (3.8)
\end{aligned}$$

Since $s^k_{(i)}$ is a solution of the Problem (3.2), we have that

$$\langle\varphi_f(x^{k,i-1},\lambda_k)+\frac{1}{2}B_{(i)}(x^{k,i-1})s^k_{(i)},s^k_{(i)}\rangle+h(x^{k,i})-h(x^{k,i-1})+\frac{\sigma_k}{2}\|s^k_{(i)}\|^2\leq 0.$$

Replacing the last inequalities in (3.8), we have

$$\begin{aligned}
&f(x^{k,i})+h(x^{k,i})\\
&\leq f(x^{k,i-1})+h(x^{k,i-1})+\|\nabla_{(i)}f(x^{k,i-1})-g_{(i)}(x^{k,i-1},\lambda_k)\|\|s^k_{(i)}\|\\
&\quad+\frac{L}{2}\|s^k_{(i)}\|^2+\frac{M}{\beta+1}\|s^k_{(i)}\|^{\beta+1}+\frac{1}{2}\|B_{(i)}(x^{k,i-1})\|\|s^k_{(i)}\|^2-\frac{\sigma_k}{2}\|s^k_{(i)}\|^2\\
&\leq f(x^{k,i-1})+h(x^{k,i-1})+\sqrt{n}\frac{L}{2}\lambda_k\|s^k_{(i)}\|+\sqrt{n}\frac{M}{\beta+1}\lambda_k^\beta\|s^k_{(i)}\|+\frac{L}{2}\|s^k_{(i)}\|^2\\
&\quad+\frac{M}{\beta+1}\|s^k_{(i)}\|^{\beta+1}+\frac{1}{2}\|B_{(i)}(x^{k,i-1})\|\|s^k_{(i)}\|^2-\frac{\sigma_k}{2}\|s^k_{(i)}\|^2.
\end{aligned}$$

By Assumption 2, this implies that

$$\begin{aligned}
F(x^{k,i})\leq&F(x^{k,i-1})+\sqrt{n}\left(\frac{L}{2}\lambda_k+\frac{M}{\beta+1}\lambda_k^\beta\right)\|s^k_{(i)}\|+\left(\frac{L}{2}+\frac{1}{2}\overline{B}\right)\|s^k_{(i)}\|^2\\
&+\frac{M}{\beta+1}\|s^k_{(i)}\|^{\beta+1}-\frac{\sigma_k}{2}\|s^k_{(i)}\|^2.
\end{aligned}$$

By previous inequality we have

$$\begin{aligned}
F(x^{k,1}) \leq & F(x^{k,0}) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)\|s_{(1)}^k\| + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)\|s_{(1)}^k\|^2 \\
& + \frac{M}{\beta+1}\|s_{(1)}^k\|^{\beta+1} - \frac{\sigma_k}{2}\|s_{(1)}^k\|^2, \\
F(x^{k,2}) \leq & F(x^{k,1}) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)\|s_{(2)}^k\| + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)\|s_{(2)}^k\|^2 \\
& + \frac{M}{\beta+1}\|s_{(2)}^k\|^{\beta+1} - \frac{\sigma_k}{2}\|s_{(2)}^k\|^2, \\
& \vdots \\
F(x^{k,q}) \leq & F(x^{k,q-1}) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)\|s_{(q)}^k\| + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)\|s_{(q)}^k\|^2 \\
& + \frac{M}{\beta+1}\|s_{(q)}^k\|^{\beta+1} - \frac{\sigma_k}{2}\|s_{(q)}^k\|^2.
\end{aligned}$$

Adding the previous inequality with i ranging from 1 to q, we have that

$$\begin{aligned}
F(x^{k,q}) \leq & F(x^{k,0}) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)\sum_{i=1}^{q}\|s_{(i)}^k\| \\
& + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)\sum_{i=1}^{q}\|s_{(i)}^k\|^2 + \frac{M}{\beta+1}\sum_{i=1}^{q}\|s_{(i)}^k\|^{\beta+1} - \frac{\sigma_k}{2}\sum_{i=1}^{q}\|s_{(i)}^k\|^2 \\
\leq & F(x^{k,0}) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)q\|s^k\|_\infty + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)q\|s^k\|_\infty^2 \\
& + \frac{M}{\beta+1}q\|s^k\|_\infty^{\beta+1} - \frac{\sigma_k}{2}\|s^k\|_\infty^2,
\end{aligned}$$

in other words

$$\begin{aligned}
F(x^{k,q}) \leq & F(x^k) + \sqrt{n}\left(\frac{L}{2}\lambda_k + \frac{M}{\beta+1}\lambda_k^{\beta}\right)q\|s^k\|_\infty + \left(\frac{L}{2} + \frac{1}{2}\overline{B}\right)q\|s^k\|_\infty^2 \\
& + \frac{M}{\beta+1}q\|s^k\|_\infty^{\beta+1} - \frac{\sigma_k}{2}\|s^k\|_\infty^2. \qquad (3.9)
\end{aligned}$$

By $\sigma_k\|s^k\|_\infty \geq \epsilon$ and $\lambda_k \leq \frac{\epsilon}{\sigma_k\sqrt{n}}$, we have $\lambda_k \leq \frac{\|s^k\|_\infty}{\sqrt{n}}$. Hence and from (3.9) we obtain

$$\begin{aligned}
F(x^{k,q}) \leq & F(x^k) + \frac{qL}{2}\|s^k\|_\infty^2 + \frac{qMn^{\frac{1-\beta}{2}}}{\beta+1}\|s^k\|_\infty^{\beta+1} + \left(\frac{L}{2} + \frac{\overline{B}}{2}\right)q\|s^k\|_\infty^2 \\
& + \frac{M}{\beta+1}q\|s^k\|_\infty^{\beta+1} - \frac{\sigma_k}{2}\|s^k\|_\infty^2 \\
\leq & F(x^k) + \frac{q(Mn^{\frac{1-\beta}{2}} + M)}{\beta+1}\|s^k\|_\infty^{\beta+1} + \left(qL + \frac{q\overline{B}}{2}\right)\|s^k\|_\infty^2 - \frac{\sigma_k}{2}\|s^k\|_\infty^2.
\end{aligned}$$

This concludes the proof of the Lemma. □

The following result (Theorem 1) establishes the existence of a $\overline{\sigma}$ such that $\sigma_k \geq \overline{\sigma}$ is sufficient to ensure the satisfiability of (3.3) in Step 4.

Theorem 1. *Let $x^{k,q}$ be obtained in Step 2 of the Algorithm 1. Suppose that Assumptions 1 and 2 hold, and $\|s^k\|_\infty \geq \epsilon/\sigma_k$. If*

$$\sigma_k \geq \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\epsilon^{\beta-1}\right]^{\frac{1}{\beta}},$$

then

$$F(x^{k,q}) \leq F(x^k) - \alpha\frac{\sigma_k}{2}\|s^k\|_\infty^2, \tag{3.10}$$

$$F(x^{k,q}) \leq F(x^k) - \frac{\alpha}{\sigma_k}\epsilon^2. \tag{3.11}$$

Proof. In Lemma 2 we obtain

$$F(x^{k,q}) \leq F(x^k) + \frac{q(Mn^{\frac{1-\beta}{2}}+M)}{\beta+1}\|s^k\|_\infty^{\beta+1} + \left(qL + \frac{q\overline{B}}{2}\right)\|s^k\|_\infty^2 - \frac{\sigma_k}{2}\|s^k\|_\infty^2.$$

Thus, taking this inequality into consideration we can conclude that to prove inequality (3.10) it is necessary to show that

$$-\frac{\alpha}{2}\sigma_k\|s^k\|^2 \geq \frac{q(Mn^{\frac{1-\beta}{2}}+M)}{\beta+1}\|s^k\|_\infty^{\beta+1} + \left(qL + \frac{q\overline{B}}{2}\right)\|s^k\|_\infty^2 - \frac{\sigma_k}{2}\|s^k\|_\infty^2,$$

which is equivalent to prove the inequality

$$\sigma_k \geq \frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\|s^k\|^{\beta-1} =: C(s^k).$$

By $\frac{\epsilon}{\sigma_k} \leq \|s^k\|$ we have that

$$\begin{aligned} C(s^k) \leq & \frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\left(\frac{\epsilon}{\sigma_k}\right)^{\beta-1} \\ \leq & \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\epsilon^{\beta-1}\right]\sigma_k^{1-\beta} \leq \sigma_k^\beta\sigma_k^{1-\beta} = \sigma_k. \end{aligned}$$

This completes the proof of (3.10). To obtain (3.11), we combine $\|s^k\|_\infty \geq \frac{\epsilon}{\sigma_k}$ with (3.10). □

This concludes the well-definition of the Algorithm 1.

The inequality (3.11) shows that choosing a very large value for σ_k may result in unnecessarily small steps. Therefore, it is recommended to start the first iteration with a small value of σ_k.

4 Convergence and complexity analysis

This section is dedicated to analyzing the complexity of Algorithm 1. Where our goal is to establish upper bounds on the number of iterations and the number of evaluations of F required to reach a previously defined optimality tolerance.

In the previous section, we demonstrated that, for σ_k sufficiently large, the reduction in F required in Step 4 of Algorithm 1 is achieved. In the following lemma, we prove that the minimum expected reduction in F corresponds to a constant factor that depends only on the constants provided by the Algorithm and Assumptions 1 and 2.

Theorem 2. *Suppose Assumptions 1 and 2 are satisfied, and let x^{k+1} be let x^{k+1} be as in Algorithm 1. Then we have that*

$$F(x^{k+1}) \leq F(x^k) - \frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}, \tag{4.1}$$

where $c = 2\max\left\{\sigma_{\min}, \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\right]^{\frac{1}{\beta}}\right\}$.

Proof. In the Theorem 1 we show that

$$\sigma_k \geq \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\epsilon^{\beta-1}\right]^{\frac{1}{\beta}},$$

then a decrease in the value of F is achieved.

In other words, when σ_k is taken satisfying large, then it is no longer necessary to increase σ_k. We can, therefore, conclude that σ_k is upper bounded by

$$\sigma_{\max} = 2\epsilon^{\frac{\beta-1}{\beta}}\max\left\{\sigma_{\min}, \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\right]^{\frac{1}{\beta}}\right\}.$$

This implies that $\sigma_{\max} = \epsilon^{\frac{\beta-1}{\beta}}c$. By replacing $\sigma_{\max}$ in (3.10), we conclude that

$$F(x^{k+1}) \quad \leq \quad F(x^k) - \frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}.$$

This completes the proof of (4.1). □

A relevant consideration is that, when the constants L, M, and β are known, we can fix the parameter σ_k as $\sigma_k = \left[\frac{2L+q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}}+M)}{(\beta+1)(1-\alpha)}\epsilon^{\beta-1}\right]^{\frac{1}{\beta}}$, which eliminates the need to update it or perform the decrement test, as the stopping criterion will always be satisfied.

By fixing σ_k in this manner, we ensure that the algorithm converges to the optimal solution without the need to adjust the regularization parameter or perform additional tests. However, it is important to note that this approach

requires prior knowledge of the constants L, M, and β, which may not always be feasible.

The following theorem shows that the stopping criterion can be achieved within a limited number of iterations. Specifically, the number of iterations required to reach the stopping criterion is bounded by a multiple of $\epsilon^{-\frac{\beta+1}{\beta}}$. This bound depends on the value of ϵ, which determines the desired level of accuracy. As ϵ decreases or $\beta \approx 0$, the number of iterations may increase. However, the bound also indicates a tradeoff between step size and convergence rate, which can be optimized to minimize the number of iterations needed.

Theorem 3. *Suppose the assumptions of Theorem 2 are fulfilled. Then, the number of iterations required to reach the stopping criterion set in Algorithm 1 is upper bounded by*

$$\frac{1}{\alpha} c\big(F(x^0) - F_{target}\big)\, \epsilon^{-\frac{\beta+1}{\beta}},$$

where c is the same as in Theorem 2.

Proof. In Theorem 2, we prove that if Algorithm 1 does not reach the established stopping criterion by iteration k, then the condition

$$F(x^j) \leq F(x^{j-1}) - \frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}$$

hold for every $j = 1, 2, \dots k$. From which it follows that

$$\begin{aligned} F_{target} <& F(x^{k-1}) - \frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}, \quad F_{target} < F(x^{k-2}) - 2\frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}, \\ & \cdots, \end{aligned}$$

i.e.,

$$F_{g_{target}} < F(x^0) - k\frac{\alpha}{c}\epsilon^{\frac{\beta+1}{\beta}}.$$

This inequality implies that

$$k \leq \frac{c\big(F(x^0) - F_{target}\big)}{\alpha}\frac{\alpha}{c}\epsilon^{-\frac{\beta+1}{\beta}}.$$

This concludes the demonstration. □

The Algorithm 1 involves evaluating various aspects at each iteration to test the stopping criterion in Step 3. Additionally, the function F must be evaluated for different test points in each iteration. Thus, it is crucial to analyze the number of times F needs to be evaluated, which corresponds to the number of times the regularization parameter σ_k needs to be increased.

Fortunately, Theorem 1 establishes that increasing the parameter σ_k is not always necessary, provided that the following condition holds:

$$\sigma_k \geq \left[\frac{2L + q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}} + M)}{(\beta+1)(1-\alpha)}\epsilon^{\beta-1}\right]^{\frac{1}{\beta}}.$$

Thus, we can conclude that the parameter σ_k is updated only a finite number of times. In the following theorem, we provide an upper bound for the number of functional evaluations required.

Theorem 4. *Suppose that the conditions of Theorem 2 hold. Then the Algorithm 1 uses at most the following number of evaluations of F and its subdifferential:*

$$\frac{c\big(F(x^0) - F_{target}\big)}{\alpha}\epsilon^{-\frac{\beta+1}{\beta}} + \log_2\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right),$$

where

$$\sigma_{\max} = 2\epsilon^{\frac{\beta-1}{\beta}} \max\left\{\sigma_{\min}, \left[\frac{2L + q\overline{B}}{(1-\alpha)} + \frac{2q(Mn^{\frac{1-\beta}{2}} + M)}{(\beta+1)(1-\alpha)}\right]^{\frac{1}{\beta}}\right\}.$$

Proof. Since $\sigma_k \leq \sigma_{\max}$, the parameter $\sigma_{\min}$ is increased a finite number of times. Let r be the maximum number of times that $\sigma_{\min}$ is increased. Then, we have:

$$2^r\sigma_{\min} \leq \sigma_{\max}, \quad \text{which is equivalent to} \quad 2^r \leq \sigma_{\max}/\sigma_{\min}.$$

This implies that:

$$r = \log_2(2^r) \leq \log_2\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right).$$

From Theorem 3, we know that the maximum number of iterations is upper bounded by:

$$\frac{c\big(F(x^0) - F_{target}\big)}{\alpha}\epsilon^{-\frac{\beta+1}{\beta}}.$$

Therefore, the number of evaluations of f and its subdifferentials employed by Algorithm 1 is upper bounded by:

$$\frac{c\big(F(x^0) - F_{target}\big)}{\alpha}\epsilon^{-\frac{\beta+1}{\beta}} + \log_2\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right).$$

This completes the proof. □

We conclude that the proposed method presents a complexity compatible with those found in the optimization literature in traditional methods, considering Lipschitz or Hölder type regularity hypotheses for the gradient of the objective function.

5 Numerical examples

In this section, we apply our proposed method to solve some problems where both f were considered with a particular case of

$$f(x) = \frac{1}{2}\|A(x) - b\|_2^2 + \frac{\lambda}{p}\|\Phi(x)\|_p^p, \tag{5.1}$$

with $1 < p < 2$, A is differentiable and Φ is linear.

In the following examples, we consider $B_{(i)} \equiv 0$, $n = 10$, $\alpha = 0.5$, $\epsilon = 5 \times 10^{-5}$, $F_{\text{target}} = 0.5$ and the function $\varphi_f : \mathbb{R}^{10} \times \mathbb{R} \to \mathbb{R}^{10}$ defined by

$$\varphi_f(x, \lambda) = \Big[\frac{f(x + \lambda e_1) - f(x)}{\lambda}, \cdots, \frac{f(x + \lambda e_{10}) - f(x)}{\lambda}\Big].$$

Example 1. In this example, we illustrate our method for solving an academic problem using different values for the initial regularization parameter, i.e., using different values for σ_0.

It is important to note that the inequality in (3.11) shows that choosing a very large value for σ_0 can result in very small decreases in the objective function, thus requiring a large number of iterations to reach the desired approximate solution. This is illustrated in the following numerical example.

We consider the following linear least squares problem, penalized by a norm L_p, with $1 < p < 2$.

$$\min F(x) = \frac{1}{2}\|Ax - b\|_2^2 + \frac{\lambda}{p}\|\Phi(x)\|_p^p, \tag{5.2}$$

where

$$A = \begin{bmatrix} 1 & 2 & 0 & 0 & 1 & 1 & 2 & 0 & 1 & 0 \\ 2 & 0 & 1 & 1 & 2 & 2 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 2 & 1 & 1 & 0 & 1 & 2 & 1 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 2 & 2 & 1 \\ 2 & 1 & 0 & 0 & 1 & 1 & 2 & 1 & 0 & 2 \\ 0 & 0 & 2 & 1 & 2 & 2 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 1 & 1 & 2 & 1 & 2 & 2 & 2 & 1 \\ 2 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 2 \\ 1 & 1 & 1 & 0 & 2 & 2 & 1 & 0 & 1 & 1 \end{bmatrix}, \; b = (1, 1, 1, 1, 1, 1, 1, 1, 1, 2).$$

In the algorithm we use $f(x) = 0.5\|Ax - b\|^2 + 5 \times 10^{-5} \sum_{i=1}^{10} |x_i|^{3/2}$ and $h(x) = 0$ and x_0 the null vector of $\mathbb{R}^{10}$.

In Table 1, we report the number of blocks (N. Blocks), value of σ_0, number of iterations (N. Iterations), and the function values at termination (Approximate F^*).

Table 1. Behavior of Algorithm BCDC-Dfree for different σ_0.

λ_k	N. Blocks	σ_0	N. Iterations	Approximate F^*
$\frac{\epsilon}{\sigma_k\sqrt{10}}$	10	1	10	0.4731311217358947
	10	500	40	0.30211898002262805
	10	1000	83	0.30027806919970645

Figure 1 illustrates the behavior of the method in the scenarios presented in Table 1.

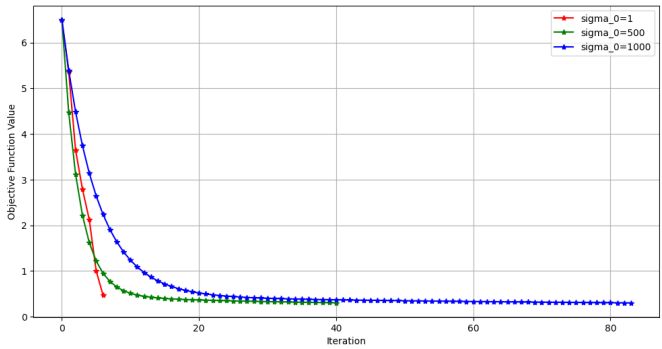

Figure 1. Evolution of the Algorithm BCDC-Dfree for different values of σ_0.

Example 2. In this example, we consider the Problem (1.1) with f defined according to Equation (5.1), where $A \in \mathbb{R}^{10\times 10}$, $b \in \mathbb{R}^{10}$ are randomly generated, Φ is the identity function and h is a null function. In this case, the problem considered is the same as (5.2), with the difference that A and b were taken randomly. We considered 100 distinct instances, all with x^0 generated randomly, and compared the approach using 10 blocks with that based on a single block. Table 2 presents the number of blocks used in the method, the initial parameter value σ_0, the average number of iterations, and the average values of the objective function at the end of the runs (Average-F). In Table 3, the only change with respect to the test in Table 2 is that σ_0 is randomly chosen within the interval $[1, 1000]$ (`rand()` $* 999 + 1$), while the other data from the previous case were kept unchanged.

Table 2. BCDC-Dfree for Example 1 with random A and b.

Blocks	σ_0	Average Iterations	Average-F
10	1	11.55	0.4329204306366679
1	1	17.56	0.4682053114984191

Table 3. BCDC-Dfree for Example 1 with random A, σ_0 and b.

Blocks	σ_0	Average Iterations	Average-F
10	$rand() * 999 + 1$	805.45	0.4987648957425285
1	$rand() * 999 + 1$	809.02	0.498824397642592

Example 3. In this example, we consider the case where A in (5.1) is nonlinear. We assume $A(x) = (A_1(x), \ldots, A_n(x))$, where $A_i(x) = \frac{x_i}{1+x_{i+1}^2}$ for $i = 1, \ldots, n-1$, with $x_{n+1} := x_1$, and $\Phi(x) = Bx$, with $B \in \mathbb{R}^{m \times n}$.

In the tests, we considered B and b generated randomly, with B containing entries composed of 0 and 1. A total of 100 distinct instances were considered, all with x^0 generated randomly. Additionally, we assumed σ_0 to be randomly chosen in the interval $[1, 100]$ (`rand()` $* 99 + 1$). Table 4 shows the number of blocks used, the average number of iterations performed, and the average values of the objective function at the end of the runs (Average-F), obtained from 100 randomly generated instances.

Table 4. BCDC-Dfree for case where A in 5.1 is nonlinear.

Blocks	σ_0	Average Iterations	Average-F
10	$rand() * 99 + 1$	648.96	0.500757739540928
1	$rand() * 99 + 1$	664.61	0.5008468045728981

6 Conclusions

We present a new block coordinate descent method for solving composite optimization problems, where the objective function combines a gradient term that satisfies a Lipschitz–Hölder condition and an additional convex term. The proposed method was designed to operate without the use of derivatives, constituting an adaptation of the proximal method. The analysis performed ensured the convergence of the algorithm under the hypothesis of cyclic selection of coordinates (or blocks of coordinates), providing clear bounds on complexity in the worst case. These results reinforce the robustness and applicability of the method in different optimization contexts.

Additionally, we include numerical examples that validate the practical efficiency of the algorithm in academic situations, demonstrating its viability for real problems. As future directions, we highlight the possibility of investigating variations of the algorithm, including random coordinate selection strategies, as well as its application in broader and larger-scale problems. These extensions can further expand the scope and efficiency of the proposed approach.

Acknowledgements

The author thanks the editor and reviewers for their valuable comments and observations.

References

[1] V.S. Amaral, R. Andreani, E.J.G. Birgin, D.S. Marcondes and J.M. Martínez. On complexity and convergence of high-order coordinate descent algorithms for

smooth nonconvex box-constrained minimizations. *J. Global Optim.*, **84**(3):527–561, 2022. https://doi.org/10.1007/s10898-022-01168-6.

[2] A. Beck and L. Tetruashvili. On the convergence of block coordinate descent type methods. *SIAM J. Optim.*, **23**(4):2037–2060, 2013. https://doi.org/10.1137/120887679.

[3] A. Bernigaud, S. Gratton and E. Simon. A non-linear conjugate gradient in dual space for L_p-norm regularized non-linear least squares with application in data assimilation. *Numer. Algorithms*, **95**:471–497, 2024. https://doi.org/10.1007/s11075-023-01578-x.

[4] E.G. Birgin and J.M. Martínez. Block coordinate descent for smooth nonconvex constrained minimization. *Computational Optimization and Applications*, **83**:1–27, 2022. https://doi.org/10.1007/s10589-022-00389-5.

[5] F. Chorobura and I. Necoara. Random coordinate descent methods for nonseparable composite optimization. *SIAM J. Optim.*, **33**(3):2160–2190, 2023. https://doi.org/10.1137/22M148700X.

[6] G.N. Grapiglia. Quadratic regularization methods with finite-difference gradient approximations. *Comput. Optim. Appl.*, **85**:683–703, 2022. https://doi.org/10.1007/s10589-022-00373-z.

[7] G.N. Grapiglia. Worst-case evaluation complexity of a derivative-free quadratic regularization method. *Optim Lett*, **18**:195–213, 2024. https://doi.org/10.1007/s11590-023-01984-z.

[8] R. Lopes, S.A. Santos and P.J.S. Silva. Accelerating block coordinate descent methods with identification strategies. *Comput. Optim. Appl.*, **72**:609–640, 2019. https://doi.org/10.1007/s10589-018-00056-8.

[9] L.F. Maia and D.H. Gutman. The randomized block coordinate descent method in the hölder smooth setting. *Optim Lett*, 2024. https://doi.org/10.1007/s11590-024-02161-6.

[10] J.M. Martínez. On high-order model regularization for constrained optimization. *SIAM J. Optim.*, **27**(4):2447–2458, 2017. https://doi.org/10.1137/17M1115472.

[11] S.J. Wright. Coordinate descent algorithms. *Mathematical Programming*, **151**(1):3–34, 2015. https://doi.org/10.1007/s10107-015-0892-3.

[12] M. Yashtini. On the global convergence rate of the gradient descent method for functions with Hölder continuous gradients. *Optim. Lett.*, **10**(6):1361–1370, 2016. https://doi.org/10.1007/s11590-015-0936-x.