Quadratic Regularization Methods with Finite-Difference Gradient Approximations

Geovani Nunes Grapiglia*

November 1, 2021

Abstract

This paper presents two quadratic regularization methods with finite-difference gradient approximations for smooth unconstrained optimization problems. One method is based on forward finite-difference gradients, while the other is based on central finite-difference gradients. In both methods, the accuracy of the gradient approximations and the regularization parameter in the quadratic models are jointly adjusted using a nonmonotone acceptance condition for the trial points. When the objective function is bounded from below and has Lipschitz continuous gradient, it is shown that the method based on forward finite-difference gradients needs at most $\mathcal{O}\left(n\epsilon^{-2}\right)$ function evaluations to generate a ϵ-approximate stationary point, where n is the problem dimension. Under the additional assumption that the Hessian of the objective is Lipschitz continuous, an evaluation complexity bound of the same order is proved for the method based on central finite-difference gradients. Numerical results are also presented. They confirm the theoretical findings and illustrate the relative efficiency of the proposed methods.

Keywords: Nonconvex Optimization; Derivative-Free Methods; Finite-Differences; Worst-Case Complexity

1 Introduction

Consider the unconstrained optimization problem

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

where $f:\mathbb{R}^n\to\mathbb{R}$ is continuously differentiable, and potentially nonconvex. Usually, problems of the form (1) are solved using iterative methods that rely on the gradient of the objective function. However, in many applications, gradient vectors are not readily available. When only function values are provided by the user, derivative-free optimization (DFO) methods are required. This class of methods includes direct-search methods, model-based methods and also methods based on finite-difference gradient approximations (see, e.g., the survey paper [13]).

In the theoretical analysis of derivative-free methods, it is of particular interest the derivation of worst-case evaluation complexity bounds. Given a *deterministic* DFO method, a worst-case evaluation complexity bound is an upper bound for the number of function evaluations required by the method to generate x_k such that

$$\|\nabla f(x_k)\|\leq\epsilon. \tag{2}$$

*Université catholique de Louvain, ICTEAM/INMA, Avenue Georges Lemaître, 4-6/ L4.05.01, B-1348, Louvain-la-Neuve, Belgium (geovani.grapiglia@uclouvain.be).

Usually, the worst-case complexity analysis of DFO methods is done assuming that the objective function has Lipschitz continuous gradients. In the context of direct-search methods, Vicente [19] and Konecny and Richtárik [12] obtained evaluation complexity bounds of $\mathcal{O}\left(n^2\epsilon^{-2}\right)$, while Dodangeh, Vicente and Zhang [5] showed that the factor n^2 is optimal for direct search methods. Regarding model-based DFO methods, Grapiglia, Yuan and Yuan [9] obtained an evaluation complexity bound of $\mathcal{O}\left(n^2|\log(\epsilon)|\epsilon^{-2}\right)$ for a derivative-free trust-region method. This bound was improved to $\mathcal{O}\left(n^2\epsilon^{-2}\right)$ by Garmanjani, Júdice and Vicente [6].

In the context of *randomized* DFO methods, Nesterov and Spokoiny [17] presented a method that needs at most $\mathcal{O}\left(n\epsilon^{-2}\right)$ function evaluations to generate x_k such that

$$E\left[\|\nabla f(x_k)\|\right] \leq \epsilon, \tag{3}$$

where $E\left[X\right]$ denotes the expected value of the random variable X. An evaluation complexity bound of the same order was obtained by Bergou, Gorbunov and Richtárik [4] for a stochastic three point method. Gratton *et al.* [10] proposed a direct-search method based on probabilistic descent that needs at most $\mathcal{O}\left(mn\epsilon^{-2}\right)$ function evaluations to generate x_k such that

$$P\left(\|\nabla f(x_k)\| \leq \epsilon\right) \geq 1 - e^{-c\epsilon^{-2}}, \tag{4}$$

where $P(A)$ denotes the probability of event A, c is a positive constant, and m is the number of random polling directions, which is defined by the user. A bound of the same order was obtained by Kimiaei and Neumaier [11] for their VSBBO algorithm with m random directions used at each iteration. More recently, Cartis and Roberts [1] presented a randomized subspace DFO method for nonlinear least-squares problems that needs at most $\mathcal{O}\left(r\epsilon^{-2}\right)$ function evaluations to generate x_k for which (4) holds, where r is the dimension of the subspaces.

In this paper, two quadratic regularization methods with finite-difference gradients are proposed for problem (1). The first method is based on forward finite-difference gradients, while the second method is based on central finite-difference gradients. In both methods, the accuracy of the gradient approximations and the regularization parameter in the quadratic models are jointly adjusted using a nonmonotone acceptance condition for the trial points. This technique was recently proposed by Grapiglia, Gonçalves and Silva [7] in the context of a cubic regularization method with finite-difference Hessians. Here, it is shown that the method based on forward finite-difference gradients needs at most $\mathcal{O}\left(n\epsilon^{-2}\right)$ function evaluations to generate x_k for which (2) holds. Under the additional assumption that the Hessian of the objective is Lipschitz continuous, an evaluation complexity bound of the same order is proved for the method based on central finite-difference gradients. Numerical results are also presented. They confirm the theoretical findings and illustrate the relative efficiency of the proposed methods.

The paper is organized as follows. Section 2 contains the basic auxiliary results. Section 3 presents the method based on forward finite-differences and its evaluation complexity analysis. Section 4 deals with the method based on central finite-differences. Finally, in Section 5, numerical results are reported.

2 Auxiliary Results

The problem class considered in this work is specified by the following assumptions:

A1. The gradient f is L_1-Lipschitz continuous, i.e.,

$$\|\nabla f(y) - \nabla f(x)\| \leq L_1 \|y - x\|, \quad \forall x, y \in \mathbb{R}^n.$$

A2. There exists $f_{low} \in \mathbb{R}$ such that $f(x) \geq f_{low}$ for all $x \in \mathbb{R}^n$.

Given $x \in \mathbb{R}^n$ and $\sigma > 0$, let $M_{x,\sigma} : \mathbb{R}^n \to \mathbb{R}$ be the quadratic model defined by

$$M_{x,\sigma}(y) = f(x) + \langle g, y - x \rangle + \frac{1}{2}\langle B(y - x), y - x \rangle + \frac{\sigma}{2}\|y - x\|^2 \tag{5}$$

where $B \in \mathbb{R}^{n \times n}$ is a symmetric positive semidefinite matrix, and $g \in \mathbb{R}^n$ is an approximation to $\nabla f(x)$.

The next lemma establishes the inequality that will define the acceptance condition for the trial points in the proposed methods.

Lemma 2.1. *Suppose that A1 holds and assume that x^+ satisfies*

$$M_{x,\sigma}(x^+) \leq f(x), \tag{6}$$

for some $x \in \mathbb{R}^n$ and $\sigma > 0$. Moreover, suppose that for some $\kappa_g \geq 0$ and $\hat{x} \in \mathbb{R}^n$ we have

$$\|\nabla f(x) - g\| \leq \kappa_g \|x - \hat{x}\|. \tag{7}$$

If

$$\sigma \geq 2\left(L_1 + \|B\| + 2\kappa_g\right), \tag{8}$$

then

$$f(x) - f(x^+) \geq \frac{\sigma}{4}\|x^+ - x\|^2 - \kappa_g \|x - \hat{x}\|^2. \tag{9}$$

Proof. By A1, (5), (6) and (7), we have

$$\begin{aligned}
f(x^+) &\leq f(x) + \langle \nabla f(x), x^+ - x \rangle + \frac{L_1}{2}\|x^+ - x\|^2 \\
&= f(x) + \langle g, x^+ - x \rangle + \frac{1}{2}\langle B(x^+ - x), x^+ - x \rangle + \frac{\sigma}{2}\|x^+ - x\|^2 \\
&\quad + \langle \nabla f(x) - g, x^+ - x \rangle - \frac{1}{2}\langle B(x^+ - x), x^+ - x \rangle + \frac{(L_1 - \sigma)}{2}\|x^+ - x\|^2 \\
&= M_{x,\sigma}(x^+) + \langle \nabla f(x) - g, x^+ - x \rangle - \frac{1}{2}\langle B(x^+ - x), x^+ - x \rangle \\
&\quad + \frac{(L_1 - \sigma)}{2}\|x^+ - x\|^2 \\
&\leq f(x) + \|\nabla f(x) - g\|\|x^+ - x\| + \frac{(\|B\| + L_1 - \sigma)}{2}\|x^+ - x\|^2 \\
&\leq f(x) + \kappa_g \|x - \hat{x}\|\|x^+ - x\| + \frac{(\|B\| + L_1 - \sigma)}{2}\|x^+ - x\|^2 \\
&\leq f(x) + \kappa_g \|x - \hat{x}\|^2 + \kappa_g \|x^+ - x\|^2 + \frac{(\|B\| + L_1 - \sigma)}{2}\|x^+ - x\|^2 \\
&= f(x) + \frac{(2\kappa_g + \|B\| + L_1 - \sigma)}{2}\|x^+ - x\|^2 + \kappa_g \|x - \hat{x}\|^2.
\end{aligned}$$

Consequently,

$$f(x) - f(x^+) \geq \frac{\sigma - (L_1 + \|B\| + 2\kappa_g)}{2}\|x^+ - x\|^2 - \kappa_g\|x - \hat{x}\|^2. \tag{10}$$

It follows from (8) that

$$-(L_1 + \|B\| + 2\kappa_g) \geq -\frac{\sigma}{2}. \tag{11}$$

Thus, combining (10) and (11) we get (9). □ □

The next lemma provides an upper bound for $\|\nabla f(x^+)\|$ when x^+ is an approximate stationary point of $M_{x,\sigma}(\cdot)$ and g is a suitable approximation of $\nabla f(x)$.

Lemma 2.2. *Suppose that A1 holds and assume that $x^+ \in \mathbb{R}^n$ satisfies*

$$\|\nabla M_{x,\sigma}(x^+)\| \leq \theta\|x^+ - x\|, \tag{12}$$

for some $x \in \mathbb{R}^n$, $\sigma > 0$ and $\theta \geq 0$. Moreover, suppose that for some $C \geq 0$ and $\hat{x} \in \mathbb{R}^n$, we have

$$\|\nabla f(x) - g\| \leq C\|x - \hat{x}\|. \tag{13}$$

Then

$$\|\nabla f(x^+)\| \leq (L_1 + \|B\| + \theta + C + \sigma)\max\left\{\|x - \hat{x}\|, \|x^+ - x\|\right\}. \tag{14}$$

Proof. By (5), A1, (12) and (13), we get

$$\begin{aligned}
\|\nabla f(x^+)\| &\leq \|\nabla f(x^+) - \nabla M_{x,\sigma}(x^+)\| + \|\nabla M_{x,\sigma}(x^+)\| \\
&= \|\nabla f(x^+) - \big(g + B(x^+ - x) + \sigma(x^+ - x)\big)\| + \|\nabla M_{x,\sigma}(x^+)\| \\
&\leq \|\nabla f(x^+) - g\| + (\|B\| + \sigma)\|x^+ - x\| + \|\nabla M_{x,\sigma}(x^+)\| \\
&\leq \|\nabla f(x^+) - \nabla f(x)\| + \|\nabla f(x) - g\| + (\|B\| + \sigma)\|x^+ - x\| \\
&\quad + \|\nabla M_{x,\sigma}(x^+)\| \\
&\leq L_1\|x^+ - x\| + C\|x - \hat{x}\| + (\|B\| + \sigma + \theta)\|x^+ - x\| \\
&\leq (L_1 + C + \|B\| + \sigma + \theta)\max\left\{\|x - \hat{x}\|, \|x^+ - x\|\right\}.
\end{aligned}$$

Therefore, (14) is true. □ □

The next lemma gives the classical upper bound for $\|\nabla f(x) - g\|$ when g is a forward finite-difference approximation of $\nabla f(x)$.

Lemma 2.3. *Suppose that A1 holds. Given $x \in \mathbb{R}^n$ and $h > 0$, let $g \in \mathbb{R}^n$ be defined by*

$$g_i = \frac{f(x + he_i) - f(x)}{h}, \quad i = 1, \ldots, m. \tag{15}$$

Then

$$\|\nabla f(x) - g\| \leq \frac{\sqrt{n}L_1}{2}h. \tag{16}$$

Combining Lemmas 2.1 and 2.3, we have the following result.

Lemma 2.4. *Suppose that A1 holds and assume that x^+ satisfies (6) for some $x \in \mathbb{R}^n$ and $\sigma > 0$. Moreover, suppose that the vector g in $M_{x,\sigma}(\,\cdot\,)$ is defined by (15) with*

$$0 < h \leq \frac{2\kappa_g}{\sqrt{n}\sigma}\|x - \hat{x}\| \tag{17}$$

for some $\kappa_g > 0$ and $\hat{x} \in \mathbb{R}^n$. If

$$\sigma \geq 2\left(L_1 + \|B\| + 2\kappa_g\right), \tag{18}$$

then

$$f(x) - f(x^+) \geq \frac{\sigma}{4}\|x^+ - x\|^2 - \kappa_g\|x - \hat{x}\|^2. \tag{19}$$

Proof. By (17) and (18) we have

$$0 < h \leq \frac{2\kappa_g}{\sqrt{n}L_1}\|x - \hat{x}\|.$$

Then, it follows from Lemma 2.3 that

$$\|\nabla f(x) - g\| \leq \kappa_g\|x - \hat{x}\|. \tag{20}$$

Finally, in view of (6), (18), (20) and Lemma 2.1, we conclude that (19) holds. □ □

The final auxiliary result provides a convergence rate for a certain sequence of nonnegative numbers.

Lemma 2.5. *Given $\tau, \lambda > 0$ and a set $\{z_j\}_{j=1}^k$ of nonnegative real numbers, with $k \geq 2$, let*

$$m(k) := \text{argmin}_{j \in \{1,\ldots,k-1\}}\left(z_j^\tau + z_{j+1}^\tau\right).$$

If $\sum_{j=1}^k z_j^\tau \leq \lambda$ then

$$\max\left\{z_{m(k)}, z_{m(k)+1}\right\} \leq \left(\frac{2\lambda}{k-1}\right)^{\frac{1}{\tau}}.$$

Proof. See Lemma 4 in [7]. □ □

3 Method based on Forward Finite-Differences

Consider now the following Quadratic Regularization Method (QRM) with forward finite-difference gradient approximations.

Algorithm 1. QRM with Forward Finite-Difference Gradients

Step 0. Given $x_0, x_1 \in \mathbb{R}^n$ ($x_0 \neq x_1$), a symmetric positive semidefinite matrix $B_1 \in \mathbb{R}^{n \times n}$, $\sigma_1 > 0$, and $\theta \geq 0$, set $\kappa_g = \sigma_1/4$, and $k := 1$.

Step 1. Find the smallest integer $i \geq 0$ such that $2^i\sigma_k \geq 2\sigma_1$.

Step 1.1. For

$$h_i = \frac{2\kappa_g\|x_k - x_{k-1}\|}{\sqrt{n}\left(2^i\sigma_k\right)}, \tag{21}$$

compute $g_{k,i} \in \mathbb{R}^n$ by

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k)}{h_i}, \quad j = 1, \ldots, n. \tag{22}$$

Step 1.2. Consider the quadratic model

$$M_{x_k, 2^i \sigma_k}(y) := f(x_k) + \langle g_{k,i}, y - x_k \rangle + \frac{1}{2} \langle B_k(y - x_k), y - x_k \rangle + \frac{2^i \sigma_k}{2} \|y - x_k\|^2,$$

and compute an approximate solution $x_{k,i}^+$ of the subproblem

$$\min_{y \in \mathbb{R}^n} M_{x_k, 2^i \sigma_k}(y), \tag{23}$$

such that

$$M_{x_k, 2^i \sigma_k}(x_{k,i}^+) \leq f(x_k) \quad \text{and} \quad \|\nabla M_{x_k, 2^i \sigma_k}(x_{k,i}^+)\| \leq \theta \|x_{k,i}^+ - x_k\|. \tag{24}$$

Step 1.3. If

$$f(x_k) - f(x_{k,i}^+) \geq \frac{2^i \sigma_k}{4} \|x_{k,i}^+ - x_k\|^2 - \frac{\sigma_1}{4} \|x_k - x_{k-1}\|^2 \tag{25}$$

holds, set $i_k = i$, $g_k = g_{k,i_k}$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2. Set $x_{k+1} = x_{k,i_k}^+$, $\sigma_{k+1} = 2^{i_k - 1} \sigma_k$, choose a symmetric positive semidefinite matrix $B_{k+1} \in \mathbb{R}^{n \times n}$, set $k := k + 1$, and go to Step 1.

Remark 3.1. *In view of (25), it is possible the acceptance of x_{k+1} such that $f(x_{k+1}) > f(x_k)$. Thus, when $B_k = I$ (for all k) and $\theta = 0$, Step 1 of Algorithm 1 can be viewed as a nonmonotone line-search procedure.*

The worst-case complexity analysis of Algorithm 1 will be done under the following additional assumption:

A3. There exists $M \geq 0$ such that $\|B_k\| \leq M$.

The next lemma establishes bounds on the sequence $\{\sigma_k\}$.

Lemma 3.2. *Suppose that A1 and A3 hold. Then, the sequence of regularization parameters $\{\sigma_k\}$ in Algorithm 1 satisfies*

$$\sigma_1 \leq \sigma_k \leq 2\left(L_1 + M + \frac{\sigma_1}{2}\right) \equiv \sigma_{\max}, \tag{26}$$

for all $k \geq 1$.

Proof. Clearly (26) is true for $k = 1$. Suppose that (26) holds for some $k \geq 1$. If $i_k = 0$, then by Step 1 we have $\sigma_{k+1} = \frac{1}{2}\sigma_k \geq \sigma_1$ and, by the induction assumption, $\sigma_{k+1} = \frac{1}{2}\sigma_k < \sigma_k \leq 2\left(L_1 + M + \frac{\sigma_1}{2}\right)$, that is, (20) holds for $k + 1$. On the other hand, if $i_k \geq 1$. Then, by Step 1 we have

$$\sigma_{k+1} = 2^{i_k - 1} \sigma_k = \frac{1}{2}\left(2^{i_k} \sigma_k\right) \geq \frac{1}{2}(2\sigma_1) = \sigma_1.$$

Now, it remains to show that

$$\sigma_{k+1} = 2^{i_k - 1} \sigma_k \leq 2\left(L_1 + M + \frac{\sigma_1}{2}\right). \tag{27}$$

Assuming by contradiction that (27) is not true, and using $\kappa_g = \sigma_1/4$, it follows from A3 that

$$2^{i_k-1}\sigma_k > 2\left(L_1 + M + \frac{\sigma_1}{2}\right) \geq 2\left(L_1 + \|B_k\| + 2\kappa_g\right).$$

In this case, by Lemma 2.4, inequality (25) would have been satisfied for $i = i_k - 1$, contradicting the minimality of i_k. Thus, (27) holds. Therefore, (26) also holds for $k+1$, which completes the induction argument. □ □

The next theorem gives an iteration complexity bound of $\mathcal{O}\left(\epsilon^{-2}\right)$ for Algorithm 1.

Theorem 3.3. *Suppose that A1-A3 hold. Given $\epsilon > 0$, let $\{x_k\}_{k=1}^{T}$ be generated by Algorithm 1 such that*

$$\|\nabla f(x_k)\| > \epsilon \quad k = 1, \ldots, T. \tag{28}$$

Then

$$T \leq 3 + (4L_1 + 3M + \sigma_1)^2 \left[\frac{8\left(f(x_1) - f_{low}\right)}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2}. \tag{29}$$

Proof. Notice that $2^{i_k}\sigma_k = 2\sigma_{k+1}$ and, by Lemma 3.2, $\sigma_k \geq \sigma_1$ for all $k \geq 1$. Then, it follows from (25) that

$$f(x_k) - f(x_{k+1}) \geq \frac{\sigma_{k+1}}{2}\|x_{k+1} - x_k\|^2 - \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2, \quad k = 1, \ldots, T-1.$$

Summing up these inequalities, it follows from A2 and $\sigma_k \geq \sigma_1$ that

$$\begin{aligned}
f(x_1) - f_{low} &\geq f(x_1) - f(x_T) = \sum_{k=1}^{T-1} f(x_k) - f(x_{k+1}) \\
&\geq \sum_{k=1}^{T-1} \frac{\sigma_{k+1}}{2}\|x_{k+1} - x_k\|^2 - \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2 \\
&= \sum_{k=1}^{T-1} \frac{\sigma_{k+1}}{2}\|x_{k+1} - x_k\|^2 - \sum_{k=1}^{T-1} \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2 \\
&= \sum_{k=2}^{T} \frac{\sigma_k}{2}\|x_k - x_{k-1}\|^2 - \sum_{k=2}^{T-1} \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2 - \frac{\sigma_1}{4}\|x_1 - x_0\|^2 \\
&\geq \sum_{k=2}^{T} \frac{\sigma_k}{2}\|x_k - x_{k-1}\|^2 - \sum_{k=2}^{T} \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2 - \frac{\sigma_1}{4}\|x_1 - x_0\|^2 \\
&= \sum_{k=2}^{T} \frac{\sigma_k}{4}\|x_k - x_{k-1}\|^2 - \frac{\sigma_1}{4}\|x_1 - x_0\|^2 \\
&\geq \frac{\sigma_1}{4}\sum_{k=1}^{T-1} \|x_{k+1} - x_k\|^2 - \frac{\sigma_1}{4}\|x_1 - x_0\|^2
\end{aligned}$$

and so

$$\sum_{k=1}^{T-1} \|x_{k+1} - x_k\|^2 \leq \frac{4(f(x_1) - f_{low})}{\sigma_1} + \|x_1 - x_0\|^2. \tag{30}$$

Let us denote $s_k = x_{k+1} - x_k$. In this way, we can write (30) as

$$\sum_{k=1}^{T-1} \|s_k\|^2 \leq \frac{4(f(x_1) - f_{low})}{\sigma_1} + \|s_0\|^2. \tag{31}$$

If $T < 3$, then (29) is satisfied. Thus, assume that $T \geq 3$ and let

$$t = \text{argmin}_{j \in \{1,\ldots,T-2\}} \left(\|s_j\|^2 + \|s_{j+1}\|^2\right).$$

Then, by Lemma 2.5 with $z_j = \|s_j\|$, $k = T-1$, $\tau = 2$ and

$$\lambda = \frac{4(f(x_1) - f_{low})}{\sigma_1} + \|s_0\|^2,$$

it follows from (31) that

$$\max\{\|s_t\|, \|s_{t+1}\|\} \leq \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|s_0\|^2\right]^{\frac{1}{2}} \frac{1}{(T-2)^{\frac{1}{2}}}. \tag{32}$$

Moreover, by (21), Lemma 3.2 and $\kappa_g = \sigma_1/4$, we have

$$\|\nabla f(x_{t+1}) - g_{t+1}\| \leq \frac{\sqrt{n}L_1}{2} \frac{2\kappa_g \|x_{t+1} - x_t\|}{\sqrt{n}\,(2\sigma_1)} \leq \frac{\kappa_g L_1}{2\sigma_1} \|x_{t+1} - x_t\| \leq \frac{L_1}{8} \|x_{t+1} - x_t\|.$$

Consequently, by (28), A3 and Lemma 2.2 with $C = L_1/8$, we have

$$\epsilon < \|\nabla f(x_{t+2})\| \leq \left(L_1 + M + \theta + \frac{L_1}{8} + \sigma_{\max}\right) \max\{\|x_{t+2} - x_{t+1}\|, \|x_{t+1} - x_t\|\},$$

which gives

$$\max\{\|s_t\|, \|s_{t+1}\|\} \geq \frac{\epsilon}{\left(L_1 + M + \sigma_{\max} + \theta + \frac{L_1}{8}\right)}. \tag{33}$$

Then, combining (32) and (33), it follows that

$$\frac{\epsilon}{\left(L_1 + M + \sigma_{\max} + \theta + \frac{L_1}{8}\right)} \leq \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]^{\frac{1}{2}} \frac{1}{(T-2)^{\frac{1}{2}}}$$

$$\Longrightarrow T - 2 \leq \left(L_1 + M + \sigma_{\max} + \frac{L_1}{8}\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right] \epsilon^{-2}.$$

Finally, using the definition of $\sigma_{\max}$ given in (26), we conclude that (29) is true. □ □

Now, taking into account the function evaluations required to obtain the approximated gradients, it is possible to establish an evaluation complexity bound of $\mathcal{O}\left(n\epsilon^{-2}\right)$ for Algorithm 1.

Corollary 3.4. *Suppose that A1-A3 hold and let $\{x_k\}_{k\geq 1}$ be generated by Algorithm 1. Given $\epsilon \in (0,1)$, assume that $T(\epsilon)$ is the first iteration index such that $\|\nabla f(x_{T(\epsilon)+1})\| \leq \epsilon$, and let $FE(\epsilon)$ be the total number of function evaluations up to the $T(\epsilon)$th iteration of Algorithm 1. Then*

$$
\begin{aligned}
FE(\epsilon) \quad &\leq \quad 2(n+2)\left(3+(3L_1+2M+\sigma_1)^2\left[\frac{8(f(x_1)-f_{low})}{\sigma_1}+2\|x_1-x_0\|^2\right]\right)\epsilon^{-2} \\
&\quad +(n+2)\left[\log_2\left(2\left(L_1+M+\frac{\sigma_1}{2}\right)\right)-\log_2(\sigma_1)\right]. \qquad (34)
\end{aligned}
$$

Proof. Notice that the at the kth iteration of Algorithm 1 the number of function evaluations is bounded by $(n+2)(i_k+1)$. On the other hand,

$$\sigma_{k+1} = 2^{i_k-1}\sigma_k \Longrightarrow (n+2)(i_k+1) = (n+2)\left[2+\log_2(\sigma_{k+1})-\log_2(\sigma_k)\right].$$

Thus

$$
\begin{aligned}
FE(\epsilon) \quad &\leq \quad \sum_{k=1}^{T(\epsilon)}(n+2)(i_k+1) = (n+2)\left(2T(\epsilon)+\log_2(\sigma_{T(\epsilon)+1})-\log_2(\sigma_1)\right) \\
&\leq \quad (n+2)\left[2T(\epsilon)+\log_2\left(2(L_1+M+\frac{\sigma_1}{2})\right)-\log_2(\sigma_1)\right]. \qquad (35)
\end{aligned}
$$

Then, combining (35), (29) and the assumption $\epsilon \in (0,1)$, we get (34). □ □

4 Method based on Central Finite-Differences

Algorithm 1 is based on forward finite-difference approximations of the gradient of $f(\cdot)$. In this section, a similar algorithm based on central finite-differences is presented. For that, consider the additional assumption:

A4. The Hessian of f is L_2-Lipschitz continuous, i.e.,

$$\|\nabla^2 f(y)-\nabla^2 f(x)\| \leq L_2\|y-x\|, \quad \forall x,y \in \mathbb{R}^n.$$

The next lemma gives the classical upper bound for $\|\nabla f(x)-g\|$ when g is a central finite-difference approximation of $\nabla f(x)$.

Lemma 4.1. *Suppose that A4 holds. Given $x \in \mathbb{R}^n$ and $h>0$, let $g \in \mathbb{R}^n$ be defined by*

$$g_i = \frac{f(x+he_i)-f(x-he_i)}{2h}, \quad i=1,\ldots,n. \qquad (36)$$

Then

$$\|\nabla f(x)-g\| \leq \frac{\sqrt{n}L_2}{6}h^2. \qquad (37)$$

Now, combining Lemmas 2.1 and 4.1 we obtain the following result. Its proof follows as the proof of Lemma 2.4.

Lemma 4.2. *Suppose that A1 and A4 hold and that x^+ satisfies (6) for some $x \in \mathbb{R}^n$ and $\sigma > 0$. Moreover, suppose that the vector g in $M_{x,\sigma}(\cdot)$ is defined by (36) with*

$$0 < h \leq \left[\frac{6\kappa_g\|x - \hat{x}\|}{\sqrt{n}\sigma}\right]^{\frac{1}{2}} \tag{38}$$

for some $\kappa_g > 0$ and $\hat{x} \in \mathbb{R}^n$. If $\sigma \geq 2\left(L_1 + L_2 + \|B\| + 2\kappa_g\right)$ then

$$f(x) - f(x^+) \geq \frac{\sigma}{4}\|x^+ - x\|^2 - \kappa_g\|x - \hat{x}\|^2. \tag{39}$$

Lemma 4.2 motivates the following Quadratic Regularization Method (QRM) with central finite-difference gradient approximations.

Algorithm 2. QRM with Central Finite-Difference Gradients

Step 0. Given $x_0, x_1 \in \mathbb{R}^n$ ($x_0 \neq x_1$), a symmetric positive semidefinite matrix $B_1 \in \mathbb{R}^{n\times n}$, $\sigma_1 > 0$, and $\theta \geq 0$, set $\kappa_g = \sigma_1/4$, and $k := 1$.

Step 1. Find the smallest integer $i \geq 0$ such that $2^i\sigma_k \geq 2\sigma_1$.

Step 1.1. For

$$h_i = \left[\frac{6\kappa_g\|x_k - x_{k-1}\|}{\sqrt{n}(2^i\sigma_k)}\right]^{\frac{1}{2}}, \tag{40}$$

compute $g_{k,i} \in \mathbb{R}^n$ by

$$[g_{k,i}]_j = \frac{f(x_k + h_i e_j) - f(x_k - h_i e_j)}{2h_i}, \quad j = 1, \ldots, n. \tag{41}$$

Step 1.2. Compute an approximate solution $x^+_{k,i}$ to the subproblem

$$\min_{y\in\mathbb{R}^n} M_{x_k,2^i\sigma_k}(y), \tag{42}$$

such that

$$M_{x_k,2^i\sigma_k}(x^+_{k,i}) \leq f(x_k) \quad \text{and} \quad \|\nabla M_{x_k,2^i\sigma_k}(x^+_{k,i})\| \leq \theta\|x^+_{k,i} - x_k\|. \tag{43}$$

Step 1.3. If

$$f(x_k) - f(x^+_{k,i}) \geq \frac{2^i\sigma_k}{4}\|x^+_{k,i} - x_t\|^2 - \frac{\sigma_1}{4}\|x_k - x_{k-1}\|^2 \tag{44}$$

holds, set $i_k = i$, $g_k = g_{k,i_k}$ and go to Step 2. Otherwise, set $i := i + 1$ and go to Step 1.1.

Step 2. Set $x_{k+1} = x^+_{k,i_k}$, $\sigma_{k+1} = 2^{i_k-1}\sigma_k$, choose a symmetric positive semidefinite matrix $B_{k+1} \in \mathbb{R}^{n\times n}$, set $k := k + 1$, and go to Step 1.

The next theorem gives an iteration complexity bound of $\mathcal{O}\left(\epsilon^{-2}\right)$ for Algorithm 2.

Theorem 4.3. *Suppose that A1-A4 hold. Given $\epsilon > 0$, let $\{x_k\}_{k\geq 0}^T$ be generated by Algorithm 2 such that*

$$\|\nabla f(x_k)\| > \epsilon, \quad k = 0, \ldots, T. \tag{45}$$

Then,

$$T \leq 3 + (3L_1 + 3L_2 + 3M + \sigma_1 + \theta)^2\left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2}. \tag{46}$$

Proof. First, let us show by induction that

$$\sigma_1 \leq \sigma_k \leq \hat{\sigma}_{\max} \equiv 2\left(L_1 + L_2 + M + \frac{\sigma_1}{2}\right) \tag{47}$$

for all $k \geq 1$. Indeed, by the definition of $\sigma_{\max}$ it is clear that (47) hods for $k = 1$. Suppose that (47) holds for some $k \geq 1$. If $i_k = 0$, then by Step 1 we have

$$\sigma_{k+1} = \frac{1}{2}\sigma_k \geq \sigma_1$$

and, by the induction assumption, $\sigma_{k+1} = \frac{1}{2}\sigma_k < \sigma_k \leq \hat{\sigma}_{\max}$, that is, (47) holds for $k+1$. On the other hand, if $i_k \geq 1$, by Step 1 we get

$$\sigma_{k+1} = 2^{i_k - 1}\sigma_k = \frac{1}{2}\left(2^{i_k}\sigma_k\right) \geq \frac{1}{2}\left(2\sigma_1\right) = \sigma_1. \tag{48}$$

Moreover, as in the proof of Lemma 3.2, it follows from Lemma 4.2, $\kappa_g = \sigma_1/4$, and A3 that

$$\sigma_{k+1} = 2^{i_k - 1}\sigma_k \leq 2\left(L_1 + L_2 + M + \frac{\sigma_1}{2}\right) = \hat{\sigma}_{\max}. \tag{49}$$

In view of (48) and (49), we see that (47) also holds for $k+1$ in this case, which completes the induction argument.

If $T < 3$, then (38) is satisfied. Thus, assume that $T \geq 3$. In this case, by following the same argument used in the proof of Theorem 3.3, it follows from (44), A2, $\sigma_k \geq \sigma_1$ and Lemma 2.5 that

$$\max\left\{\|x_{t+1} - x_t\|, \|x_{t+2} - x_{t+1}\|\right\} \leq \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]^{\frac{1}{2}} \frac{1}{(T-2)^{\frac{1}{2}}} \tag{50}$$

for

$$t = \operatorname{argmin}_{j \in \{1,\ldots,T-2\}}\left(\|x_{j+1} - x_j\|^2 + \|x_{j+2} - x_{j+1}\|^2\right).$$

Moreover, by (40) and Lemma 4.2, we have

$$\|\nabla f(x_{t+1}) - g_{t+1}\| \leq \frac{\sqrt{n}L_2}{6}\left(\left[\frac{6\kappa_g\|x_{t+1} - x_t\|}{\sqrt{n}\left(2\sigma_1\right)}\right]^{\frac{1}{2}}\right)^2 \leq \frac{L_2}{8}\|x_{t+1} - x_t\|$$

Consequently, by (45), Lemma 2.2 with $C = L_2/8$, and (47), we have

$$\epsilon < \|\nabla f(x_{t+2})\| \leq \left(L_1 + \frac{L_2}{8} + M + \hat{\sigma}_{\max} + \theta\right)\max\left\{\|x_{t+2} - x_{t+1}\|, \|x_{t+1} - x_t\|\right\},$$

which gives

$$\max\left\{\|x_{t+2} - x_{t+1}\|, \|x_{t+1} - x_t\|\right\} > \frac{\epsilon}{\left(L_1 + \frac{L_2}{8} + M + \hat{\sigma}_{\max} + \theta\right)}. \tag{51}$$

Then, combinining (50) and (51), it follows that

$$\frac{\epsilon}{\left(L_1 + \frac{L_2}{8} + M + \hat{\sigma}_{\max} + \theta\right)} \leq \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]^{\frac{1}{2}} \frac{1}{(T-2)^{\frac{1}{2}}},$$

and so

$$T - 2 \leq \left(L_1 + \frac{L_2}{8} + M + \hat{\sigma}_{\max} + \theta\right)^2 \left[\frac{8(f(x_1) - f_{low})}{\sigma_1} + 2\|x_1 - x_0\|^2\right]\epsilon^{-2}.$$

Using the definition of $\hat{\sigma}_{\max}$ in (47), we get (46). □ □

As a consequence of Theorem 2, it is possible to establish an evaluation complexity bound of $\mathcal{O}\left(n\epsilon^{-2}\right)$ for Algorithm 2.

Corollary 4.4. *Suppose that A1-A4 hold and let* $\{x_k\}_{k\geq 1}$ *be generated by Algorithm 1. Given* $\epsilon \in (0,1)$*, assume that* $T(\epsilon)$ *is the first iteration index such that* $\|\nabla f(x_{T(\epsilon)+1})\| \leq \epsilon$*, and let* $FE(\epsilon)$ *be the total number of function evaluations performed up to the* $T(\epsilon)$*th iteration of Algorithm 2. Then*

$$\begin{aligned} FE(\epsilon) &\leq 4(n+1)\left(3+(3L_1+3L_2+3M+\sigma_1+\theta)^2\left[\frac{8(f(x_1)-f_{low})}{\sigma_1}+2\|x_1-x_0\|^2\right]\right)\epsilon^{-2} \\ &\quad +2(n+1)\left[\log_2\left(2(L_1+L_2+M+\frac{\sigma_1}{2}\right)-\log_2(\sigma_1)\right]. \end{aligned} \tag{52}$$

Proof. Notice that the number of function evaluations performed at the kth iteration of Algorithm 2 is bounded by $2(n+1)(i_k+1)$. On the other hand,

$$\sigma_{k+1} = 2^{i_k-1}\sigma_k \Longrightarrow 2(n+1)(i_k+1) = 2(n+1)\left[2+\log_2(\sigma_{k+1})-\log_2(\sigma_k)\right],$$

and so

$$\begin{aligned} FE(\epsilon) &\leq \sum_{k=1}^{T(\epsilon)} 2(n+1)(i_k+1) = 2(n+2)\left(2T(\epsilon)+\log_2(\sigma_{T(\epsilon)+1})-\log_2(\sigma_1)\right) \\ &\leq 2(n+1)\left[2T(\epsilon)+\log_2\left(2(L_1+L_2+M+\frac{\sigma_1}{2}\right)-\log_2(\sigma_1)\right]. \end{aligned} \tag{53}$$

Then, combining (53), (46) and the assumption that $\epsilon \in (0,1)$, we get (52). □ □

5 Numerical Experiments

Numerical experiments were performed comparing the following Octave implementations:

- **FDGM**: Algorithm 1 with $B_k = I$ for all k.
- **FDBFGS**: Algorithm 1 with B_k computed by the BFGS update whenever it is possible, i.e.,

$$B_{k+1} = \begin{cases} B_k + \dfrac{y_k y_k^T}{s_k^T y_k} - \dfrac{B_k s_k s_k^T B_k}{s_k^T B_k s_k}, & \text{if } s_k^T y_k > 0, \\ B_k & \text{otherwise,} \end{cases} \tag{54}$$

  with $B_1 = I$, $s_k = x_{k+1} - x_k$ and $y_k = g(x_{k+1}) - g(x_k)$, where $g(x_k) = g_k$ and $g(x_{k+1})$ is the approximation to $\nabla f(x_{k+1})$ obtained by forward finite-differences with $h = h_{k,i_k}$.

- **FCBFGS**: Algorithm 2 with B_k updated by (54) where, in the definition of y_k, vector $g(x_k) = g_k$ and $g(x_{k+1})$ is the approximation to $\nabla f(x_{k+1})$ obtained by central finite-differences with $h = h_{k,i_k}$.
- **NMSMAX**: an implementation of the Nelder-Mead method [16], freely available from the Matrix Computation Toolbox[1]

[1] http://www.maths.manchester.ac.uk/~higham/mctoolbox

In codes FDGM, FDBFGS and FCBFGS, the following parameters were used

$$\sigma_1 = 10^{-2}, \quad \|x_1 - x_0\| = 10^{-3} \quad \text{and} \quad \theta = 0.$$

The experiments consisted in applying the referred codes to the set of 15 problems from the Moré-Garbow-Hillstrom collection [14] in which the dimension n can be chosen[2] . For each problem, two choices of starting points were considered, namely

$$x_0 = 5^s \bar{x}, \tag{55}$$

with $s \in \{0, 1\}$, where $\bar{x}$ is the standard starting point given in [14].

In the first experiment, code FDGM was applied to the test problems with $n = 8$ and the choice $s = 1$ for the starting points. To investigate the ability of FDGM to generate approximate stationary points, the code was endowed with the stopping criterion

$$\|\nabla f(x_k)\| \leq \epsilon. \tag{56}$$

Table 1 shows the numerical results for $\epsilon \in \left\{10^{-1}, 10^{-2}\right\}$, where $T(\epsilon)$ denotes the number of iterations required by the solver to generate x_k for which (56) holds, $FE(\epsilon)$ denotes the corresponding number of function evaluations, and $A(\epsilon)$ is defined as

$$A(\epsilon) = \frac{FE(\epsilon)}{T(\epsilon)(n+2)}. \tag{57}$$

	$\epsilon = 10^{-1}$			$\epsilon = 10^{-2}$		
PROBLEM	$T(\epsilon)$	$FE(\epsilon)$	$A(\epsilon)$	$T(\epsilon)$	$FE(\epsilon)$	$A(\epsilon)$
1. Extended Rosenbrock	5022	90540	1.8029	7422	133740	1.8019
2. Extended Powell Singular	279	5148	1.8452	886	16074	1.8142
3. Penalty I	14	324	2.3143	14	324	2.3143
4. Penalty II	16	387	2.4188	44	891	2.0250
5. Variably Dimensioned	414	7587	1.8326	605	11025	1.8223
6. Trigonometric	4	162	4.0500	28	567	2.0250
7. Discrete Boundary Value	11	297	2.7000	824	14931	1.8120
8. Discrete Integral Equation	3	126	4.2000	5	162	3.2400
9. Broyden Tridiagonal	21	504	2.4000	30	657	2.1900
10. Broyden Banded	16	405	2.5312	20	486	2.4300
11. Brown Almost Linear	17	432	2.5412	18	450	2.5000
12. Linear	4	144	3.6000	6	180	3.0000
13. Linear-1	4	279	6.9750	4	279	6.9750
14. Linear-0	10	369	3.6900	11	387	3.5182
15. Chebyquad	6	261	4.3500	8	297	3.7125

Table 1: Numerical results for FDGM.

From Table 1 we can see that solver FDGM was able to generate approximate stationary points in all problems. Figure 1 presents all the pairs $(T(\epsilon), A(\epsilon))$. Notice that as $T(\epsilon)$ grows, the corresponding number $A(\epsilon)$ becomes bounded from above by 2. This is accordance with inequality (35) established in the proof of Corollary 3.4.

[2]The MATLAB/Octave codes of the test problems are freely available in the websites `https://www.mat.univie.ac.at/~neum/glopt/test.html#test_unconstr` and
`https://people.sc.fsu.edu/~jburkardt/octave_src/test_nonlin/test_nonlin.html`.

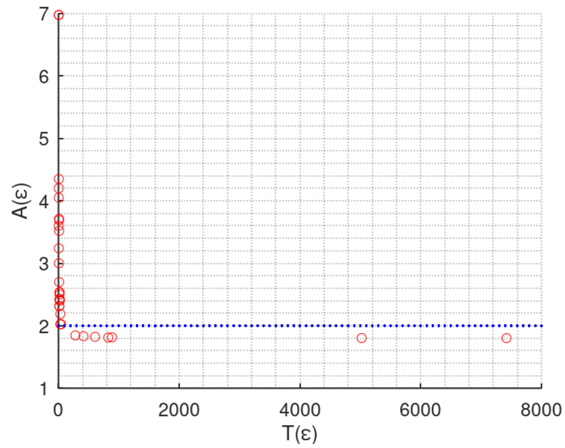

Figure 1: Pairs $(T(\epsilon), A(\epsilon))$ described in Table 1.

On the other hand, Theorem 3.3 predicts that $T(\epsilon) \leq \hat{C}\epsilon^{-2}$, where the constant $\hat{C} > 0$ depends on the problem and on the parameters used in Algorithm 1. By assuming that

$$T(\epsilon) = \hat{C}\epsilon^{-p}, \quad \epsilon > 0,$$

the power p can be numerically estimated by the formula [8]:

$$p = \frac{1}{\log(\tau)} \log\left(\frac{T(\epsilon/\tau)}{T(\epsilon)}\right), \tag{58}$$

where $\tau > 1$. Table 2 shows the estimated powers for the problems in Table 1 with respect to $\epsilon = 10^{-1}$ and $\tau = 10$. As we can see, all the estimated powers are smaller than 2. The largest power obtained was $p = 1.8745$ for problem 7. These results are in accordance with Theorem 3.3.

PROBLEM	$T(\epsilon)$	$T(\epsilon/\tau)$	p
1. Extended Rosenbrock	5022	7422	0.1696
2. Extended Powell Singular	279	886	0.5018
3. Penalty I	14	14	0.0000
4. Penalty II	16	44	0.4393
5. Variably Dimensioned	414	605	0.1648
6. Trigonometric	4	28	0.8451
7. Discrete Boundary Value	11	824	1.8745
8. Discrete Integral Equation	3	5	0.2218
9. Broyden Tridiagonal	21	30	0.1549
10. Broyden Banded	16	20	0.0969
11. Brown Almost Linear	17	18	0.0248
12. Linear	4	6	0.1761
13. Linear-1	4	4	0.0000
14. Linear-0	10	11	0.0414
15. Chebyquad	6	8	0.1249

Table 2: Numerical estimation of the complexity power p in $T(\epsilon) = \hat{C}\epsilon^{-p}$.

In the second experiment, FDGM and FDBFGS were applied to the test problems with $n = 40$. For each problem, both choices of the starting point (55) were considered ($s = 0, 1$), resulting in 30 instances. A budget of 4,100 function evaluations was allowed for the two codes (i.e, 100 simplex gradients for $n = 40$). The corresponding *data profiles* [15][3] are shown in Figure 2. It can see that code FDBFGS (a quasi-Newton version of Algorithm 1) solved more problems than FDGM using the same number of function evaluations.

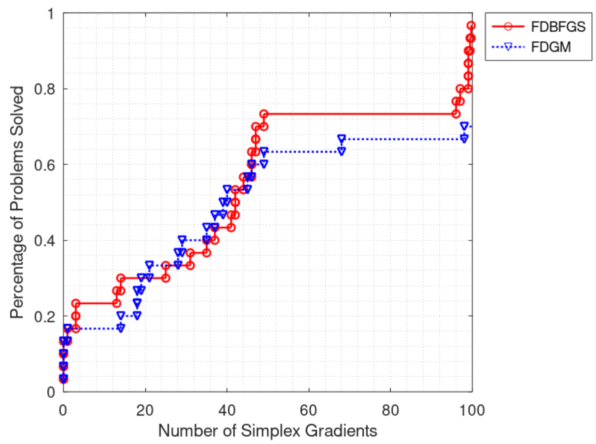

Figure 2: Data Profiles for the precision 10^{-7} and budget of 100 simplex gradients.

In the third experiment, codes FDBFGS, FCBFGS and NMSMAX were applied to the same set of 30 problems considered in the second experiment. As it can be seen in Figure 3, code FDBFGS was again the most efficient solver. Moreover, FCBFGS also outperformed NMSMAX.

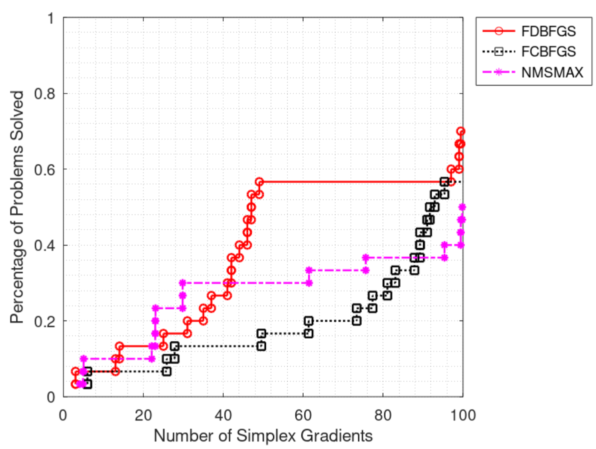

Figure 3: Data Profiles for the precision 10^{-7} and budget 100 simplex gradients.

[3]The data profiles were generated using the code **data_profile.m** freely available in the website `https://www.mcs.anl.gov/~more/dfo/`.

6 Conclusion

In this paper, quadratic regularization methods based on finite-difference gradients were proposed for nonconvex unconstrained optimization problems. When the objective function is bounded from below and has Lipschitz continuous gradients, it was shown that the method based on forward finite-difference gradients needs at most $\mathcal{O}\left(n\epsilon^{-2}\right)$ function evaluations to generate a ϵ-approximate stationary point. When, additionally, the objective has Lipschitz continuous Hessian, an evaluation complexity bound of the same order was established for the method based on central finite-differences. In both methods, the accuracy of the gradient approximations and the regularization parameter in the quadratic models are jointly adjusted using a nonmonotone acceptance condition for the trial points. The explicity use of the problem dimension in this procedure allowed the derivation of evaluation complexity bounds with linear dependence in n. Numerical results were also presented, and confirmed the theoretical findings. In particular, implementations of both methods outperformed the code NMSMAX, while the method based on forward finite-differences compared favorably with the method based on central finite-differences. As a topic of future research it would be interesting to investigate the possible adaptation of the methods proposed here to handle optimization problems with noisy functions [2, 3, 18].

References

[1] Cartis, C., Roberts, L.: Scalable Subspace Methods for Derivative-Free Nonlinear Least-Squares Optimization. arXiv:2102.12016 [math.OC] (2021)

[2] Berahas, A.S., Byrd, R.H., Nocedal, J.: Derivative-free optimization of noisy functions via quasi-Newton methods. SIAM Journal on Optimization 29, 965–993 (2019)

[3] Berahas, A.S., Cao, L., Choromanski, K., Scheinberg, K.: A theoretical and empirical comparison of gradient approximations in derivative-free optimization. Foundations of Computational Mathematics (2021)

[4] Bergou, E.H., Gorbunov, E., Richtárik, P.: Stochastic three points method for unconstrained smooth minimization. SIAM Journal on Optimization 30, 2726–2749 (2020)

[5] Dodangeh, M., Vicente, L.N., Zhang, Z.: On the optimal order of worst case complexity of direct search. Optimization Letters 10, 699–708 (2016)

[6] Garmanjani, R., Júdice, D., Vicente, L.N.: Trust-region methods without using derivatives: Worst-case complexity and the non-smooth case. SIAM Journal on Optimization 26, 1987–2011 (2016)

[7] Grapiglia, G.N., Gonçalves, M.L.N., Silva, G.N.: A Cubic Regularization of Newton's Method with Finite-Difference Hessian Approximations. Numerical Algorithms (2021), DOI: `https://doi.org/10.1007/s11075-021-01200-y`

[8] Grapiglia, G.N., Sachs, E.W.: On the worst-case evaluation complexity of non-monotone line search algorithms. Computational Optimization and Applications 68, 555-577 (2017)

[9] Grapiglia, G.N., Yuan, J., Yuan, Y.: A derivative-free trust-region algorithm for composite nonsmooth optimization. Computational and Applied Mathematics 35, 475–499 (2016)

[10] Gratton, S., Royer, C.W., Vicente, L.N., Zhang, Z.: Direct search based on probabilistic descent. SIAM Journal on Optimization 25, 1515–1541 (2015)

[11] Kimiaei, M., Neumaier, A.: Efficient global unconstrained black box optimization. Optimization Online (2021)

[12] Konecny, J., Richtárik, P.: Simple complexity analysis of simplified direct search. arXiv:1410.0390 [math.OC] (2014)

[13] Larson, J., Menickelly, M., Wild, S.M.: Derivative-Free Optimization. Acta Numerica, 287–404 (2019)

[14] Moré, J.J., Garbow, B.S., Hillstrom, K.E.: Testing unconstrained optimization software. ACM Transactions on Mathematical Software 7, 17-41 (1981)

[15] Moré, J.J., Wild, S.M.: Benchmarking derivative-Free optimization algorithms. SIAM Journal on Optimization 20, 172–191 (2009)

[16] Nelder, J.A., Mead, R.: A simplex method for function minimization. The Computer Journal 7, 308–313 (1965)

[17] Nesterov, Yu., Spokoiny, V.: Random gradient-free minimization of convex functions. Foundations of Computational Mathematics 17, 527–566 (2017)

[18] Shi, H-J.M., Xuan, M.Q., Oztoprak, F., Nocedal, J.: On the numerical performance of derivative-free optimization methods based on finite-difference approximations. arXiv:2102.09762 [math.OC] (2021)

[19] Vicente, L.N.: Worst case complexity of direct search. EURO Journal on Computational Optimization 1, 143–153 (2013)