Worst Case Complexity of Direct Search under Convexity

M. Dodangeh* L. N. Vicente†

November 18, 2014

Abstract

In this paper we prove that the broad class of direct-search methods of directional type, based on imposing sufficient decrease to accept new iterates, exhibits the same worst case complexity bound and global rate of the gradient method for the unconstrained minimization of a convex and smooth function.

More precisely, it will be shown that the number of iterations needed to reduce the norm of the gradient of the objective function below a certain threshold is at most proportional to the inverse of the threshold. It will be also shown that the absolute error in the function values decay at a sublinear rate proportional to the inverse of the iteration counter.

In addition, we prove that the sequence of absolute errors of function values and iterates converges r-linearly in the strongly convex case.

Keywords: Derivative-free optimization, direct search, worst case complexity, global rate, sufficient decrease, convexity.

1 Introduction

In this paper we focus on directional direct-search methods applied to the minimization of a real-valued, convex, and continuously differentiable objective function f, without constraints,

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

In direct-search methods, the objective function is evaluated, at each iteration, at a finite number of points. No derivatives are required. The action of declaring an iteration successful (moving into a point of lower objective function value) or unsuccessful (staying at the same iterate) is based on objective function value comparisons. Some of these methods are directional in the sense of moving along predefined directions along which the objective function will eventually decrease for sufficiently small step sizes (see, e.g., [3, Chapter 9]). Those of simplicial type (see, e.g., [3, Chapter 8]), such as the Nelder-Mead method, are not considered here. There are essentially two ways of globalizing direct-search methods (of directional type), meaning making

*Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal (dodangeh@mat.uc.pt). Support for this author was provided by FCT under the scholarship SFRH/BD/51168/2010.

†CMUC, Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal (lnv@mat.uc.pt). Support for this research was provided by FCT under grants PTDC/MAT/116736/2010 and PEst-C/MAT/UI0324/2011.

them convergent to stationary points independently of the starting point: (i) by integer lattices, insisting on generating points in grids or meshes (which refine only with the decrease of the step size), or (ii) by imposing a sufficient decrease condition, involving the size of the steps, on the acceptance of new iterates. Although we derive our results for the latter strategy, we recall that both share the essentials of this class of direct-search methods: the directional feature for the displacements, and, as in any other direct-search technique, the fact that decisions in each iteration are taken solely by comparison of objective function values.

The analyzes of global convergence of algorithms can be complemented or refined by deriving worst case complexity (WCC) bounds for the number of iterations or number of function evaluations, an information which becomes valuable in many instances. In terms of the derivation of WCC bounds, Nesterov [11, Page 29] first showed that the steepest descent or gradient method for unconstrained optimization takes at most $\mathcal{O}(\epsilon^{-2})$ iterations (or gradient evaluations) to drive the norm of the gradient of the objective function below $\epsilon \in (0, 1)$. Such a bound has been proved sharp or tight by Cartis, Gould, and Toint [1]. There has been quite an amount of research on WCC bounds for several other classes of algorithms in the non-convex case (see, e.g., [7, 9, 14]).

Derivative-free or zero-order methods have also been recently analyzed with the purpose of establishing their WCC bounds. Vicente [17] has shown a WCC bound of $\mathcal{O}(\epsilon^{-2})$ for the number of iterations of direct-search methods (of directional type, when imposing sufficient decrease, and applied to a smooth, possibly non-convex function), which translates to $\mathcal{O}(n^2\epsilon^{-2})$ in terms of the number of function evaluations. Cartis, Gould, and Toint [2] have derived a WCC bound of $\mathcal{O}(n^2\epsilon^{-3/2})$ for their adaptive cubic overestimation algorithm when using finite differences to approximate derivatives. In the non-smooth case, using smoothing techniques, both Garmanjani and Vicente [6] and Nesterov [12] established a WCC bound of approximately $\mathcal{O}(\epsilon^{-3})$ iterations for their zero-order methods, where the threshold ϵ refers now to the gradient of a smoothed version of the original function.

Nesterov [11, Section 2.1.5] has also shown that the gradient method achieves an improved WCC bound of $\mathcal{O}(\epsilon^{-1})$ if the objective function is convex. For derivative-free optimization, Nesterov [12] proved that his random Gaussian approach also attains (but now in expectation) the $\mathcal{O}(\epsilon^{-1})$ in the convex (smooth) case. It is thus natural to ask if one can achieve a similar bound for deterministic zero-order methods, and direct search offers a simple and instructive setting to answer such a question. In this paper, we will show that direct search can indeed achieve a bound of $\mathcal{O}(\epsilon^{-1})$ under the presence of convexity. The derived WCC bound measures the maximum number of iterations required to find a point where the norm of the gradient of the objective function is below ϵ, and, once again, it is proved for directional direct-search methods when a sufficient decrease condition based on the size of the steps is imposed to accept new iterates. As in the non-convex case, the corresponding maximum number of objective function evaluations becomes $\mathcal{O}(n^2\epsilon^{-1})$.

In the convex case it is also possible to derive global rates for the absolute error in function values when the solutions set is nonempty. Such an error is known to decay at a sublinear rate of $1/k$ for the gradient method when the function is convex. The rate is global since no assumption on the starting point is made. We derive in this paper a similar rate for direct search. Note that the random Gaussian approach [12] only achieves such a rate in expectation.

As in the gradient method, we also go one step further and show that the absolute error in function values as well as in the iterates converges globally and r-linearly when the function is strongly convex. Such a rate applies to the whole sequence of iterates and its derivation does not require a monotone nonincrease of the step size (as it is the case of a similar r-linear rate

derived for direct search globalized using integer lattices by Dolan, Lewis, and Torczon [4]).

Our results are derived for convex functions where the supreme distance between any point in the initial level set and the solutions set is bounded. Such property is satisfied when the solutions set is bounded (including strongly convexity as a particular case), but it is also met in several instances where the solutions sets are unbounded.

The structure of the paper is as follows. In Section 2, we briefly comment on the worst case complexity (WCC) bounds and global rates of the gradient or steepest descent method. In Section 3, we describe the class of direct search under consideration and provide the known results (global asymptotics and WCC bounds) for the smooth (continuously differentiable) and non-convex case. Then, in Section 4, we derive the global rate and WCC bound for such direct-search methods in the also smooth but now convex case. The strongly convex case is covered in Section 5. Some numerical experiments are reported in Section 6 to illustrate how the class of direct-search methods studied in this paper compares to the random Gaussian approach in [12] and the corresponding two-points improvement suggested in [5]. In Section 7 we draw some concluding remarks based on the specifics of the material covered during the paper. We note that the notation $\mathcal{O}(A)$ has meant and will mean a multiple of A, where the constant multiplying A does not depend on the iteration counter k of the method under analysis (thus depending only on f or on algorithmic constants which are set at the initialization of the method). The dependence of A on the dimension n of the problem will be made explicit whenever appropriate. The vector norms will be ℓ_2 ones. Given an open subset Ω of $\mathbb{R}^n$, we denote by $\mathcal{C}^1_\nu(\Omega)$ the set of continuously differentiable functions in Ω with Lipschitz continuous gradient in Ω, where ν is the Lipschitz constant of the gradient. We use the notation $\mathcal{F}(\Omega)$ to represent the set of convex functions defined on a convex set Ω. The intersection of both is denoted by $\mathcal{F}^1_\nu(\Omega) = \mathcal{F}(\Omega) \cap \mathcal{C}^1_\nu(\Omega)$, where Ω is open and convex. Finally, since our functions will always be assumed continuously differentiable, uniform and strong convexity are thus equivalent notions and from now on we will talk only about strongly convex functions. The notation $\mathcal{F}^1_{\nu,\mu}(\Omega)$ will denote the subset of $\mathcal{C}^1_\nu(\Omega)$ formed by the strongly convex functions with constant $\mu > 0$.

2 WCC of gradient-type methods

Given a starting point $x_0 \in \mathbb{R}^n$, the gradient or steepest descent method takes the form $x_{k+1} = x_k - h_k \nabla f(x_k)$, where $h_k > 0$ defines the step size. The algorithm can be applied whenever the function f is continuously differentiable, and the well known fact that $-\nabla f(x_k)$ is a descent direction provides the basis for the convergence properties of the method. The update of the step size h_k is also a crucial point in this class of minimization algorithms. There are improper choices of the step size that make such gradient-type algorithms diverge [15, Chapter 3]. The proper update of the step size is thus central in achieving global convergence (see, e.g., [11, 15]).

For a number of the well known strategies to update the step size, it is possible to prove that, when $f \in \mathcal{C}^1_\nu(\mathbb{R}^n)$, there is a constant $C = C(\nu) > 0$ such

$$f(x_k) - f(x_{k+1}) \;\geq\; C(\nu)\|\nabla f(x_k)\|^2, \tag{2}$$

where $C(\nu)$ is essentially a multiple of $1/\nu$, with ν the Lipschitz constant of the gradient of f, (being the multiple dependent on the parameters involved in the update of the step size; in [11, Page 29], for instance, $C(\nu) = 1/(2\nu)$ for $h_k = 1/\nu$). In such cases, assuming that f is also bounded from below on $\mathbb{R}^n$, one can show that the gradient method takes at most $\mathcal{O}(\epsilon^{-2})$

iterations to reduce the norm of the gradient below $\epsilon > 0$ (see [11, Page 29]), to be more specific

$$\left(\frac{f(x_0) - f_{low}}{C(\nu)}\right)\frac{1}{\epsilon^2}.$$

The constant multiplying ϵ^{-2} depends thus only on ν, on the parameters involved in the update of the step size, on and the lower bound f_{low} for f in $L_f(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$.

If, additionally, f is assumed convex, i.e., $f \in \mathcal{F}^1_\nu(\mathbb{R}^n)$, then Nesterov [11, Section 2.1.5] showed that one can achieve a better WCC bound in terms of the negative power of ϵ. First, based on the geometric properties of smooth convex functions (essentially [11, Equation (2.1.7)]), he proved, for simplicity using $h_k = 1/\nu$, that the absolute error in function values decays at a sublinear rate of $1/k$

$$f(x_k) - f_* \;\leq\; \frac{2\nu\|x_0 - x_*\|^2}{k+4}, \tag{3}$$

where f_* is the value of the function at a (global) minimizer (see [11, Corollary 2.1.2]), assumed to exist. But then one can easily see, by repeatedly applying (2), that

$$\begin{aligned}
\frac{2\nu\|x_k - x_*\|}{k+2} &\geq f(x_k) - f_* \\
&\geq C(\nu)\|\nabla f(x_k)\|^2 + f(x_{k+1}) - f_* \\
&\geq C(\nu)\sum_{\ell=k}^{2k}\|\nabla f(x_\ell)\|^2 + f(x_{2k+1}) - f_* \\
&\geq C(\nu)\sum_{\ell=k}^{2k}\|\nabla f(x_\ell)\|^2.
\end{aligned}$$

Again, $C(\nu) = 1/(2\nu)$ when $h_k = 1/\nu$. The gradient method is then proved to only take at most $\mathcal{O}(\epsilon^{-1})$ iterations to achieve a threshold of ϵ on the norm of the gradient. The constant multiplying ϵ^{-1} is essentially a multiple of $\nu\|x_0 - x_*\|$.

3 WCC of direct search

The direct-search method under analysis is described in Algorithm 3.1, following the presentation in [3, Chapter 7]. The directional feature is presented in the poll step, where points of the form $x_k + \alpha_k d$, for directions d belonging to the positive spanning set D_k, are tested for sufficient decrease (and by a positive spanning set we mean a set of nonzero directions that spans $\mathbb{R}^n$ with non-negative coefficients). For this purpose, following the terminology in [10], $\rho : (0,\infty) \to (0,\infty)$ will represent a forcing function, i.e., a non-decreasing (typically continuous) function satisfying $\lim_{t\downarrow 0}\rho(t)/t = 0$. Typical examples of forcing functions are $\rho(t) = \mathcal{C}t^p$, for $p > 1$ and $\mathcal{C} > 0$. The poll step is successful if the value of the objective function is sufficiently decreased relatively to the step size α_k, in the sense that there exists $d_k \in D_k$ such that $f(x_k + \alpha_k d_k) < f(x_k) - \rho(\alpha_k)$, in which case the step size is possibly increased. Failure in doing so defines an unsuccessful iteration, and the step size is decreased by a factor strictly less than 1 that changes between two bounds which need to be fixed during the course of the iterations. The search step is purposely left open since it does not interfere in any of the convergence or complexity properties of the algorithm, and it is solely used to improve the practical performance of the overall algorithm.

Algorithm 3.1 (Directional direct-search method)

Initialization

Choose x_0 with $f(x_0) < +\infty$, $\alpha_0 > 0$, $0 < \beta_1 \leq \beta_2 < 1$, and $\gamma \geq 1$.

For $k = 0, 1, 2, \ldots$

1. **Search step:** Try to compute a point with $f(x) < f(x_k) - \rho(\alpha_k)$ by evaluating the function f at a finite number of points. If such a point is found, then set $x_{k+1} = x$, declare the iteration and the search step successful, and skip the poll step.
2. **Poll step:** Choose a positive spanning set D_k. Order the set of poll points $P_k = \{x_k + \alpha_k d : d \in D_k\}$. Evaluate f at the poll points following the chosen order. If a poll point $x_k + \alpha_k d_k$ is found such that $f(x_k + \alpha_k d_k) < f(x_k) - \rho(\alpha_k)$, then stop polling, set $x_{k+1} = x_k + \alpha_k d_k$, and declare the iteration and the poll step successful. Otherwise, declare the iteration (and the poll step) unsuccessful and set $x_{k+1} = x_k$.
3. **Mesh parameter update:** If the iteration was successful, then maintain or increase the step size parameter: $\alpha_{k+1} \in [\alpha_k, \gamma\alpha_k]$. Otherwise, decrease the step size parameter: $\alpha_{k+1} \in [\beta_1\alpha_k, \beta_2\alpha_k]$.

In the poll step one may move opportunistically to the first point where sufficient decrease was found or continue for complete polling where the point with the lowest function value is then chosen.

When the objective function is bounded from below one can prove that there exists a subsequence of unsuccessful iterates driving the step size parameter to zero (see [10] or [3, Theorems 7.1 and 7.11 and Corollary 7.2]).

Lemma 3.1 *Let f be bounded from below on $L_f(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$. Then Algorithm 3.1 generates an infinite subsequence K of unsuccessful iterates for which* $\lim_{k \in K} \alpha_k = 0$.

Note that when the function f is convex and has a minimizer, it is necessarily bounded from below.

To continue towards the global properties (asymptotic convergence and rates) for this class of direct search, one must look at the key feature of a positive spanning set, its cosine measure [10]. Given a positive spanning set D, its cosine measure is given by

$$\mathrm{cm}(D) \;=\; \min_{0 \neq v \in \mathbb{R}^n} \max_{d \in D} \frac{v^\top d}{\|v\|\|d\|}.$$

Any positive spanning set has a positive cosine measure. This fact means for any non-zero vector, in particular the negative gradient at a given point, there is at least one direction in D making an acute angle with it. Such a property enables us to derive that the norm of the gradient is of the order of the step size when an unsuccessful iteration occurs [4, 10] (see also [3, Theorem 2.4 and Equation (7.14)]).

Theorem 3.1 *([4, 10]) Let D_k be a positive spanning set and $\alpha_k > 0$ be given. Assume that ∇f is Lipschitz continuous (with constant $\nu > 0$) in an open set containing all the poll points in P_k. If $f(x_k) \leq f(x_k + \alpha_k d) + \rho(\alpha_k)$, for all $d \in D_k$, i.e., the iteration k is unsuccessful, then*

$$\|\nabla f(x_k)\| \ \leq \ \mathrm{cm}(D_k)^{-1} \left(\frac{\nu}{2} \alpha_k \max_{d \in D_k} \|d\| + \frac{\rho(\alpha_k)}{\alpha_k \min_{d \in D_k} \|d\|} \right). \tag{4}$$

It becomes then obvious that one needs to avoid degenerate positive spanning sets.

Assumption 3.1 *All positive spanning sets D_k used for polling (for all k) must satisfy $\mathrm{cm}(D_k) \geq \mathrm{cm}_{\min}$ and $d_{\min} \leq \|d\| \leq d_{max}$ for all $d \in D_k$ (where $\mathrm{cm}_{\min} > 0$ and $0 < d_{\min} \leq d_{\max}$ are constants).*

A first global asymptotic result is then easily obtained by combining Lemma 3.1 and Theorem 3.1 (under Assumption 3.1), and ensures the convergence to zero of the gradient at a subsequence of unsuccessful iterates. Moreover, we have the following WCC bounds in this general non-convex, smooth setting [17].

Theorem 3.2 *([17]) Consider the application of Algorithm 3.1 when $\rho(\alpha) = \mathcal{C}\alpha^p$, $p > 1$, $\mathcal{C} > 0$, and D_k satisfies Assumption 3.1. Let f be bounded from below in $L_f(x_0)$ and $f \in \mathcal{C}^1_\nu(\Omega)$ where Ω is an open set containing $L_f(x_0)$.*

Under these assumptions, to reduce the norm of the gradient below $\epsilon \in (0, 1)$, Algorithm 3.1 takes at most

$$\mathcal{O}((\sqrt{n}\nu\epsilon^{-1})^{\hat{p}}),$$

iterations, and at most

$$\mathcal{O}(n(\sqrt{n}\nu\epsilon^{-1})^{\hat{p}}).$$

function evaluations, where $\hat{p} = \frac{p}{\min(1,p-1)}$.

When $p = 2$, these numbers are of $\mathcal{O}(n\nu^2\epsilon^{-2})$ and $\mathcal{O}(n^2\nu^2\epsilon^{-2})$, respectively.

The constant in $\mathcal{O}(\cdot)$ depends only on $d_{\min}$, $d_{\max}$, $\mathrm{cm}_{\min}$, $\mathcal{C}$, p, β_1, β_2, γ, α_0, and on the lower bound of f in $L_f(x_0)$.

How the step size α_k is updated impacts in several ways the WCC bounds given above for Algorithm 3.1. In fact, the choice of $\mathcal{C}$ in the forcing function and the choice of the parameters β_1, β_2, and γ in the step size updating formulas influence the constant in the bound (4). Increasing $\mathcal{C}$, for instance, will decrease the number of successful iterations [17, Theorem 3.1], possibly leading to more unnecessary unsuccessful iterations and consequently more unnecessary function evaluations. Increasing the value of the expansion factor $\gamma \geq 1$ will increase the maximum number of unsuccessful iterations compared to the number of successful ones [17, Theorem 3.2], again possibly leading to more unnecessary unsuccessful iterations and consequently more unnecessary function evaluations. Setting $\gamma = 1$ leads to an optimal choice in this respect. One practical strategy to accommodate $\gamma > 1$ is by considering an upper bound for the step size itself.

Assumption 3.2 *There is a positive constant M such that $\alpha_k \leq M$ for $\forall k \geq 0$.*

This is not a strong assumption at all since one can always pick a constant $M > \alpha_0$ and then update the step size at successful iterations by $\alpha_{k+1} \in [\alpha_k, \min\{\gamma\alpha_k, M\}]$. Under this assumption Theorem 3.1 simplifies to the following:

Corollary 3.1 *Consider $\rho(\alpha_k) = \mathcal{C}\alpha_k^p$, $p > 1$, $\mathcal{C} > 0$. Under the assumptions of Theorem 3.1 and Assumptions 3.1 and 3.2, if $f(x_k) \leq f(x_k + \alpha_k d) + \rho(\alpha_k)$, for all $d \in D_k$, i.e., the iteration k is unsuccessful, then*

$$\|\nabla f(x_k)\| \ \leq \ \mathrm{cm}_{\min}^{-1} \ \frac{\frac{\nu}{2} d_{\max} M + \mathcal{C} d_{\min}^{-1} M^{p-1}}{M^{\min(1,p-1)}} {\alpha_k}^{\min(1,p-1)}. \tag{5}$$

The step size upper bound M will appear thus in the upper bound for the gradient in unsuccessful iterations. When $p = 2$, the upper bound on the gradient does not depends on M,

$$\|\nabla f(x_k)\| \ \leq \ \mathrm{cm}_{\min}^{-1} \left(\frac{\nu}{2} d_{\max} + \mathcal{C} d_{\min}^{-1} \right) \alpha_k.$$

The analysis of worst case complexity for the convex case when $p \neq 2$ will, however, depend on the upper bound M for the step size.

4 WCC of direct search for a class of convex functions

In this section, we will analyze the WCC of direct search when the objective function is smooth and convex under the following assumption.

Assumption 4.1 *The solutions set $X_*^f = \{x_* \in \mathbb{R}^n : x_* \text{ is a minimizer of } f\}$ for problem (1) is nonempty. The level set $L_f(x) = \{y \in \mathbb{R}^n : f(y) \leq f(x)\}$ is bounded for some x or, if that is not the case, $\sup_{y \in L_f(x_0)} \mathrm{dist}(y, X_*^f)$ is still finite.*

If $L_f(x_0)$ is bounded, then $\sup_{y \in L_f(x_0)} \mathrm{dist}(y, X_*^f)$ is trivially finite.

Furthermore, it is known that when a convex function f is proper and closed (meaning semi-continuous) the following property is true (see [16, Corollary 8.7.1]): if $\{x \in \mathbb{R}^n : f(x) \leq \alpha\}$ is nonempty and bounded for some α, then it is bounded for all α. In particular, since we assume that X_*^f is nonempty, $X_*^f = L_f(x_*)$ for some x_*, and if X_*^f is bounded so is $L_f(x_0)$. Moreover, a (finite, thus continuous) strongly convex function in $\mathbb{R}^n$ has a unique minimizer x_*, which then makes X_*^f nonempty and bounded.

In conclusion, and generally speaking, strong convexity of f and boundedness of either X_*^f or $L_f(x_0)$ fulfill the above assumption and make $\sup_{y \in L_f(x_0)} \mathrm{dist}(y, X_*^f)$ finite. Let

$$R \ = \ \sup_{y \in L_f(x_0)} \mathrm{dist}(y, X_*^f).$$

Note that there are convex functions f such that $\sup_{y \in L_f(x_0)} \mathrm{dist}(y, X_*^f)$ is finite but neither f is strongly convex nor $L_f(x)$ is bounded for any x, being such an instance the two-dimensional function $f(x, y) = y^2$. There are however some rare pathological instances where Assumption 4.1 does not hold (see the end of the paper).

Note also that assuming the finiteness of the longest distance from the initial level set to the solutions set is unnecessary in the gradient method since it can be proved that for a constant step

size smaller than $2/\nu$ the iterates satisfy $\|x_k - x_*\| \leq \|x_0 - x_*\|$ (see Nesterov [11, Theorem 2.1.13]). The lack of knowledge of the gradient makes the control of the longest distance to the solutions set harder in direct search.

To avoid repeating the several assumptions in the statements of the results of this section we will combine them in the following one.

Assumption 4.2 *Consider the application of Algorithm 3.1 when $\rho(t) = \mathcal{C}\, t^p$, $p > 1$, $\mathcal{C} > 0$, and D_k satisfies Assumption 3.1. Let $f \in \mathcal{F}_\nu^1(\Omega)$, where Ω is an open and convex set containing $L_f(x_0)$. Let Assumption 3.2 (when $p \neq 2$) and Assumption 4.1 also hold.*

We will make extensive use of the sets $\mathcal{S}(k_0, j)$ and $\mathcal{U}(k_0, j)$ to represent the indices of successful and unsuccessful iterations, respectively, between k_0 (including it) and j (excluding it).

4.1 Global rate on function values

We will start by measuring the decrease obtained in the objective function until a given iteration as a function of the number of successful iterations occurred until then. Recall that $f_* = f(x_*)$ for some $x_* \in X_*^f$ and $\hat{p} = \frac{p}{\min(1,p-1)} \geq 2$ for $p > 1$.

Lemma 4.1 *Let Assumptions 4.2 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence $\{x_k\}_{k \geq k_0}$ such that*

$$(f(x_k) - f_*)^{\hat{p}-1} \;<\; \frac{R^{\hat{p}}}{\omega |\mathcal{S}(k_0, k)|}, \tag{6}$$

where

$$\omega \;=\; \omega_g^{\hat{p}} \beta_1^p \mathcal{C}, \quad \omega_g \;=\; \frac{2\, \mathrm{cm}_{\min}\, M^{\min(1,p-1)}}{\nu d_{\max} M + 2\mathcal{C} d_{\min}^{-1} M^{p-1}}, \tag{7}$$

and $|\mathcal{S}(k_0, k)|$ is the number of successful iterations between k_0 (including it) and k.

Proof. Let $\mathcal{U}(k_0, k) = \{k_i\}_{i=0}^m$ represent the set of unsuccessful iterations which occur between iteration k_0, inclusively, and iteration k. One has $|\mathcal{S}(k_0, k)| = k - k_0 - m - 1$. Since all iterations between k_m and k are successful and k_m is unsuccessful, we have that

$$\begin{aligned}
f(x_k) &< f(x_{k-1}) - \mathcal{C}\alpha_{k-1}^p \\
&\;\;\vdots \\
&< f(x_{k_m+1}) - \mathcal{C} \sum_{j=k_m+1}^{k-1} \alpha_j^p \\
&\leq f(x_{k_m+1}) - \mathcal{C}(k - k_m - 1)\alpha_{k_m+1}^p \\
&\leq f(x_{k_m}) - \beta_1^p \mathcal{C}(k - k_m - 1)\alpha_{k_m}^p.
\end{aligned}$$

Now, by Corollary 3.1,

$$f(x_k) \;<\; f(x_{k_m}) - (k - k_m - 1)\omega \|\nabla f(x_{k_m})\|^{\hat{p}}. \tag{8}$$

By applying a similar argument, but now starting from x_{k_i}, $i = m, \ldots, 1$, we deduce that

$$f(x_{k_i}) \;<\; f(x_{k_{i-1}}) - (k_i - k_{i-1} - 1)\omega \|\nabla f(x_{k_{i-1}})\|^{\hat{p}}. \tag{9}$$

Denote $\Delta f_i = f(x_{k_i}) - f_*$, for $i = 0, \ldots, m$ and $\Delta f_{m+1} = f(x_k) - f_*$. Then, using the property stated in [11, Equation (2.1.7)] for $f \in \mathcal{F}^1_\nu(\Omega)$,

$$\begin{aligned} f_* &= f(x^i_*) \\ &\geq f(x_{k_i}) + \langle \nabla f(x_{k_i}), x^i_* - x_{k_i} \rangle + \frac{1}{2\nu} \|\nabla f(x^i_*) - \nabla f(x_{k_i})\|^2 \\ &\geq f(x_{k_i}) + \langle \nabla f(x_{k_i}), x^i_* - x_{k_i} \rangle, \end{aligned}$$

where, for $i = 0, \ldots, m$, x^i_* is the projection of x_{k_i} onto the solutions set X^f_* (which is convex and closed since f is convex and continuous). Thus, using Assumption 4.1,

$$\begin{aligned} \Delta f_i &\leq \langle \nabla f(x_{k_i}), x_{k_i} - x^i_* \rangle \\ &\leq \|\nabla f(x_{k_i})\| \|x_{k_i} - x^i_*\| \\ &\leq R \|\nabla f(x_{k_i})\|, \quad i = 0, \ldots, m. \end{aligned} \tag{10}$$

By combining inequalities (8), (9), and (10) and setting here for simplicity $k_{m+1} = k$, we obtain, for $i = 1, \ldots, m, m+1$,

$$\Delta f_i \;<\; \Delta f_{i-1} - \frac{\omega}{R^{\hat{p}}}(k_i - k_{i-1} - 1)\Delta f^{\hat{p}}_{i-1} \;<\; \Delta f_{i-1}. \tag{11}$$

Hence, $\Delta f_{i-1}/\Delta f_i > 1$, $i = 1, \ldots, m, m+1$. Now we divide the first inequality in (11) by $\Delta f_i \Delta f_{i-1}$, then use $\hat{p} \geq 2$ and $\Delta f_{i-1} > \Delta f_{m+1}$, and later $\Delta f_{i-1}/\Delta f_i > 1$,

$$\begin{aligned} \frac{1}{\Delta f_i} &> \frac{1}{\Delta f_{i-1}} + \frac{\omega}{R^{\hat{p}}}(k_i - k_{i-1} - 1)\frac{\Delta f^{\hat{p}-1}_{i-1}}{\Delta f_i} \\ &> \frac{1}{\Delta f_{i-1}} + \frac{\omega \Delta f^{\hat{p}-2}_{m+1}}{R^{\hat{p}}}(k_i - k_{i-1} - 1)\frac{\Delta f_{i-1}}{\Delta f_i} \\ &> \frac{1}{\Delta f_{i-1}} + \frac{\omega \Delta f^{\hat{p}-2}_{m+1}}{R^{\hat{p}}}(k_i - k_{i-1} - 1). \end{aligned} \tag{12}$$

By summing the inequality (12) for $i = 1, \ldots, m, m+1$, we arrive at

$$\begin{aligned} \frac{1}{\Delta f_{m+1}} &> \frac{1}{\Delta f_0} + \frac{\omega \Delta f^{\hat{p}-2}_{m+1}}{R^{\hat{p}}}(k_{m+1} - k_0 - m - 1) \\ &> \frac{\omega \Delta f^{\hat{p}-2}_{m+1}}{R^{\hat{p}}}(k_{m+1} - k_0 - m - 1), \end{aligned}$$

or, equivalently,

$$\begin{aligned} (f(x_k) - f_*)^{\hat{p}-1} &= \Delta f^{\hat{p}-1}_{m+1} \\ &< \frac{R^{\hat{p}}}{\omega(k_{m+1} - k_0 - m - 1)} \\ &= \frac{R^{\hat{p}}}{\omega(k - k_0 - m - 1)}, \end{aligned}$$

as we wanted to prove (since, remember, $|\mathcal{S}(k_0,k)| = k - k_0 - m - 1$). $\square$

Following [17, Theorem 3.2] one can also guarantee that the number of unsuccessful iterations is of the same order as the number of successful ones.

Lemma 4.2 *Let Assumptions 4.2 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence $\{x_k\}_{k\geq k_0}$ such that*

$$|\mathcal{U}(k_0,k)| \;\leq\; \left\lceil \omega_1|\mathcal{S}(k_0,k)| + \omega_2 + \frac{1}{\min(1,p-1)}\log_{\beta_2}\left(\frac{\omega_g}{R}(f(x_k)-f_*)\right)\right\rceil, \tag{13}$$

where

$$\omega_1 \;=\; -\log_{\beta_2}(\gamma), \quad \omega_2 \;=\; \log_{\beta_2}(\beta_1/\alpha_{k_0}), \tag{14}$$

ω_g is given in (7), and $|\mathcal{S}(k_0,k)|$ and $|\mathcal{U}(k_0,k)|$ are the number of successful and unsuccessful iterations between k_0 (including it) and k, respectively.

Proof. Since $f \in \mathcal{F}^1_\nu(\Omega)$ and X^f_* is non-empty, one has for each unsuccessful iteration k_i (with $k_0 \leq k_i \leq k$)

$$f(x_k) - f_* \;\leq\; f(x_{k_i}) - f_* \;\leq\; \langle \nabla f(x_{k_i}), x_{k_i} - x^i_*\rangle,$$

where x^i_* is the projection of x_{k_i} onto the solutions set X^f_* (which, again, is convex and closed since f is convex and continuous). Then, from Assumption 4.1,

$$\frac{1}{R}(f(x_k) - f_*) \;\leq\; \|\nabla f(x_{k_i})\|. \tag{15}$$

From Corollary 3.1 and the definition of ω_g in (7), we have, for each unsuccessful iteration k_i, that

$$\|\nabla f(x_{k_i})\| \;\leq\; \omega_g^{-1}\alpha_{k_i}^{\min(1,p-1)}. \tag{16}$$

As before, we can backtrack from any iteration j after k_0 to the nearest unsuccessful iteration (say k_ℓ, with $k_\ell \geq k_0$) and, due to the step size updating rules, (16) implies then

$$\alpha_j \;\geq\; \beta_1(\omega_g\|\nabla f(x_{k_\ell})\|)^{\frac{1}{\min(1,p-1)}}, \quad j = k_0, k_0+1, \ldots, k$$

(which holds trivially from (16) if j is itself unsuccessful). Combining the above inequality with (15) gives a lower bound for each step size α_j

$$\alpha_j \;\geq\; \beta_1\left(\frac{\omega_g}{R}(f(x_k)-f_*)\right)^{\frac{1}{\min(1,p-1)}}, \quad j = k_0, k_0+1, \ldots, k.$$

On the other hand, one knows that either $\alpha_j \leq \beta_2\alpha_{j-1}$ or $\alpha_j \leq \gamma\alpha_{j-1}$. Hence, by induction,

$$\alpha_k \;\leq\; \alpha_{k_0}\gamma^{|\mathcal{S}(k_0,k)|}\beta_2^{|\mathcal{U}(k_0,k)|}.$$

In conclusion one has

$$\beta_1\left(\frac{\omega_g}{R}(f(x_k)-f_*)\right)^{\frac{1}{\min(1,p-1)}} \;\leq\; \alpha_k \;\leq\; \alpha_{k_0}\gamma^{|\mathcal{S}(k_0,k))|}\beta_2^{|\mathcal{U}(k_0,k)|},$$

from which we conclude,

$$f(x_k) - f_* \;\leq\; \frac{R}{\omega_g}\left(\frac{\alpha_{k_0}}{\beta_1}\gamma^{|\mathcal{S}(k_0,k)|}\beta_2^{|\mathcal{U}(k_0,k)|}\right)^{\min(1,p-1)}.$$

Now, since $\beta_2 < 1$, the function $\log_{\beta_2}(\cdot)$ is monotonically decreasing, and one obtains (the coefficient ω_1 is nonnegative due to $\gamma \geq 1$)

$$|\mathcal{U}(k_0,k)| \;\leq\; \omega_1|\mathcal{S}(k_0,k)| + \omega_2 + \frac{1}{\min(1,p-1)}\log_{\beta_2}\left(\frac{\omega_g}{R}(f(x_k)-f_*)\right).$$

□

Lemmas 4.1 and 4.2 lead to a sub-linear convergence rate for the absolute error in the function values after the first unsuccessful iteration.

Theorem 4.1 *Let Assumptions 4.2 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence $\{x_k\}_{k\geq k_0}$ such that*

$$(f(x_k)-f_*)^{\hat{p}-1} \;<\; \frac{\kappa_1}{k-\kappa_2}, \quad \forall k > \kappa_2,$$

where

$$\begin{aligned}
\kappa_1 &= (1-\log_{\beta_2}(\gamma))\frac{R^{\hat{p}}}{\omega} - \log_{\beta_2}(\exp(1)),\\
\kappa_2 &= \frac{f(x_0)-f_*}{\mathcal{C}\alpha_0^p} + \log_{\beta_2}\left(\frac{\beta_1}{\alpha_{k_0}}\right) + \frac{1}{\min(1,p-1)}\log_{\beta_2}\left(\frac{\omega_g}{R}[f(x_0)-f_*]^{1-(\hat{p}-1)\min(1,p-1)}\right),
\end{aligned}$$

and ω and ω_g are given in (7).

Proof. Due to the definition of k_0 and the step size updating rules one has

$$k_0\mathcal{C}\alpha_0^p \;\leq\; \sum_{j=0}^{k_0-1}\mathcal{C}\alpha_j^p \;<\; \sum_{j=0}^{k_0-1} f(x_j) - f(x_{j+1}) \;=\; f(x_0) - f(x_{k_0})$$

and so

$$k_0 \;<\; \frac{f(x_0)-f_*}{\mathcal{C}\alpha_0^p}. \tag{17}$$

By applying Lemmas 4.1 and 4.2 and inequality (17) one has

$$\begin{aligned}
k - \frac{f(x_0) - f_*}{\mathcal{C}\alpha_0^p} &< k - k_0 \\
&= |\mathcal{U}_k(k_0, k)| + |\mathcal{S}_k(k_0, k)| \\
&\leq (1 - \log_{\beta_2}(\gamma))|\mathcal{S}_k(k_0, k)| + \log_{\beta_2}(\beta_1/\alpha_{k_0}) \\
&\quad + \frac{1}{\min(1, p-1)} \log_{\beta_2}\left(\frac{\omega_g}{R}(f(x_k) - f_*)\right) \\
&\leq (1 - \log_{\beta_2}(\gamma))\frac{R^{\hat{p}}}{\omega}\frac{1}{(f(x_k) - f_*)^{\hat{p}-1}} + \log_{\beta_2}(\beta_1/\alpha_{k_0}) \\
&\quad + \frac{1}{\min(1, p-1)} \log_{\beta_2}\left(\frac{\omega_g}{R}(f(x_k) - f_*)\right) \\
&= (1 - \log_{\beta_2}(\gamma))\frac{R^{\hat{p}}}{\omega}\frac{1}{(f(x_k) - f_*)^{\hat{p}-1}} + \log_{\beta_2}(\beta_1/\alpha_{k_0}) \\
&\quad + \frac{1}{\min(1, p-1)} \log_{\beta_2}\left(\frac{\omega_g}{R}(f(x_{k_0}) - f_*)\right) \\
&\quad + \frac{1}{\min(1, p-1)} \log_{\beta_2}\left(\frac{f(x_k) - f_*}{f(x_{k_0}) - f_*}\right). \qquad (18)
\end{aligned}$$

Note that for any $p > 1$, one has $\hat{p} \geq 2$ and so $\hat{p} - 1 \geq 1$. Since $1/\min(1, p-1)$ is equal to $\hat{p} - 1$ when $1 < p \leq 2$ and to 1 when $p > 2$, it holds $1/\min(1, p-1) \leq \hat{p} - 1$. From $(f(x_k) - f_*)/(f(x_{k_0}) - f_*) \leq 1$ and $\beta_2 < 1$, one has $\log_{\beta_2}((f(x_k) - f_*)/(f(x_{k_0}) - f_*)) \geq 0$. So, from (18)

$$\begin{aligned}
k - \frac{f(x_0) - f_*}{\mathcal{C}\alpha_0^p} &< (1 - \log_{\beta_2}(\gamma))\frac{R^{\hat{p}}}{\omega}\frac{1}{(f(x_k) - f_*)^{\hat{p}-1}} + \log_{\beta_2}(\beta_1/\alpha_{k_0}) \\
&\quad + \frac{1}{\min(1, p-1)} \log_{\beta_2}\left(\frac{\omega_g}{R}[f(x_{k_0}) - f_*]^{1-(\hat{p}-1)\min(1,p-1)}\right) \\
&\quad + (\hat{p} - 1)\log_{\beta_2}(f(x_k) - f_*). \qquad (19)
\end{aligned}$$

Now, given $\bar{\epsilon} \in (0, \infty)$,

$$\begin{aligned}
(\hat{p} - 1)\log_{\beta_2}(\bar{\epsilon}) &= -\log_{\beta_2}(\bar{\epsilon}^{(1-\hat{p})}) \\
&= -\log_{\beta_2}(\exp(1))\ln(\bar{\epsilon}^{(1-\hat{p})}) \\
&\leq -\log_{\beta_2}(\exp(1))\bar{\epsilon}^{(1-\hat{p})}, \qquad (20)
\end{aligned}$$

where the last inequality follows from $\ln(x) \leq x$, $x > 0$.

Then, from (20) with $\bar{\epsilon} = f(x_k) - f_*$, one has

$$(\hat{p} - 1)\log_{\beta_2}(f(x_k) - f_*) \leq -\log_{\beta_2}(\exp(1))\frac{1}{(f(x_k) - f_*)^{\hat{p}-1}}$$

and the proof is concluded by plugging this inequality in (19) and using $k > \kappa_2$. □

4.2 WCC bounds

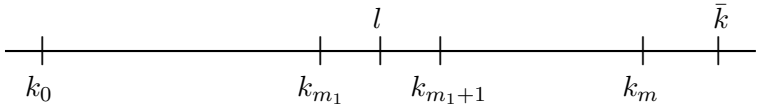
In the following lemma, by using the result of Lemma 4.1, we will derive an upper bound for the number of successful iterations after the first unsuccessful one needed to achieve an iterate for which the norm of the gradient is below a given threshold.

Lemma 4.3 *Let Assumptions 4.2 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Given any $\epsilon \in (0,1)$, assume that $\|\nabla f(x_{k_0})\| > \epsilon$ and let $\bar{k}$ be the first iteration after k_0 such that $\|\nabla f(x_{\bar{k}})\| \leq \epsilon$. Then, to achieve $\|\nabla f(x_{\bar{k}})\| \leq \epsilon$, starting from k_0, Algorithm 3.1 takes at most $|\mathcal{S}(k_0, \bar{k})|$ successful iterations, where*

$$|\mathcal{S}(k_0, \bar{k})| \;\leq\; \left\lceil 2\frac{R}{\omega}\epsilon^{1-\hat{p}} + 1 \right\rceil \tag{21}$$

and ω is given in (7).

Proof. Let l, with $k_0 < l < \bar{k}$, be the index of a successful iteration occurring before $\bar{k}$, $m+1 = |\mathcal{U}(k_0, \bar{k})|$ be number of unsuccessful iterations between k_0 (including it) and $\bar{k}$, m_1 be the number of unsuccessful iterations between k_0 and l, and $k_1, k_2, \ldots, k_m$ be the sequence of unsuccessful iterations between k_0 and $\bar{k}$.



Let us assume first that there are unsuccessful iterations between l and $\bar{k}$ (like in the figure above). Exactly as in the derivation of inequalities (8)–(9), applying also Corollary 3.1 and the step size updating rules, we have

$$f(x_{\bar{k}}) \;<\; f(x_{k_m}) - (\bar{k} - k_m - 1)\omega\|\nabla f(x_{k_m})\|^{\hat{p}}$$

and,

$$\begin{aligned} f(x_{k_i}) &< f(x_{k_{i-1}}) - (k_i - k_{i-1} - 1)\omega\|\nabla f(x_{k_{i-1}})\|^{\hat{p}}, \quad m_1 + 2 \leq i \leq m, \\ f(x_{k_{m_1+1}}) &< f(x_l) - (k_{m_1+1} - l)\omega\|\nabla f(x_{k_{m_1}})\|^{\hat{p}}. \end{aligned}$$

Summing up these inequalities and considering $\|\nabla f(x_k)\| > \epsilon$ for $k < \bar{k}$ lead us to

$$f(x_l) \;>\; f(x_{\bar{k}}) + (\bar{k} - l - m + m_1)\omega\epsilon^{\hat{p}}.$$

If there are no unsuccessful iterations between l and $\bar{k}$, $m = m_1$ and this inequality is also true by a similar argument. On the other hand, by Lemma 4.1

$$(f(x_l) - f_*)^{\hat{p}-1} \;\leq\; \frac{R^{\hat{p}}}{\omega(l - k_0 - m_1 - 1)}.$$

So, in conclusion

$$\begin{aligned} (\bar{k} - l - m + m_1)\omega\epsilon^{\hat{p}} &\leq (\bar{k} - l - m + m_1)\omega\epsilon^{\hat{p}} + f(x_{\bar{k}}) - f_* \\ &\leq f(x_l) - f_* \\ &\leq \left(\frac{R^{\hat{p}}}{\omega(l - k_0 - m_1 - 1)}\right)^{\frac{1}{\hat{p}-1}}. \end{aligned} \tag{22}$$

Now we choose l such that the number of successful iterations after l is at most one times higher than the number of successful iterations until l. To explicitly describe l we divide the number of successful iterations into two parts $(\bar{k} - k_0 - m - 1)/2$, then add the number m_1 of unsuccessful iterations until the middle point, and finally shift by k_0. Hence l is given by

$$l \;=\; \left\lfloor \frac{\bar{k} - k_0 - m - 1}{2} \right\rfloor + k_0 + m_1 + 1.$$

With such a choice of l, the number κ of successful iterations between k_0 and l is

$$\kappa \;=\; l - k_0 - m_1 - 1$$

and a simple argument shows that

$$\kappa \;=\; l - k_0 - m_1 - 1 \;\leq\; \bar{k} - l - m + m_1 \;\leq\; \kappa + 1, \tag{23}$$

as expected.

Now, from (22),

$$\begin{aligned} (\omega\kappa)^{\frac{\hat{p}}{\hat{p}-1}} &\leq \omega(\bar{k} - l - m + m_1)[\omega(l - k_0 - m_1 - 1)]^{\frac{1}{\hat{p}-1}} \\ &\leq R^{\frac{\hat{p}}{\hat{p}-1}}\epsilon^{-\hat{p}}, \end{aligned}$$

and

$$\kappa \;\leq\; \frac{R}{\omega}\epsilon^{1-\hat{p}}. \tag{24}$$

But due to equation (23), $2\kappa + 1$ is bigger than the number of successful iterations between k_0 and $\bar{k}$,

$$\begin{aligned} 2\kappa + 1 &= \kappa + 1 + \kappa \\ &\geq (\bar{k} - l - m + m_1) + (l - k_0 - m_1 - 1) \\ &= \bar{k} - k_0 - m - 1, \end{aligned}$$

which finishes the proof. □

One can also guarantee that the number of unsuccessful iterations is of the same order as the number of successful ones.

Lemma 4.4 *Let Assumptions 4.2 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Given any $\epsilon \in (0, 1)$, assume that $\|\nabla f(x_{k_0})\| > \epsilon$ and let $\bar{k}$ be the first iteration after k_0 such that $\|\nabla f(x_{\bar{k}})\| \leq \epsilon$. Then, to achieve $\|\nabla f(x_{\bar{k}})\| \leq \epsilon$, starting from k_0, Algorithm 3.1 takes at most $|\mathcal{U}(k_0, \bar{k})|$ unsuccessful iterations, where*

$$|\mathcal{U}(k_0, \bar{k})| \;\leq\; \left\lceil \omega_1 |\mathcal{S}(k_0, \bar{k})| + \omega_2 + \frac{1}{\min(p-1, 1)} \log_{\beta_2}(\omega_g \epsilon) \right\rceil,$$

ω_g is given in (7), and ω_1 and ω_2 are given in (14).

Proof. The proof is similar to the one of Lemma 4.2 using $\bar{k}-1$ instead of k and ϵ instead of $(f(x_k)-f_*)/R$. The bound will then be on $|\mathcal{U}(k_0,\bar{k}-1)|$ but $|\mathcal{U}(k_0,\bar{k})| = |\mathcal{U}(k_0,\bar{k}-1)|$ since $\bar{k}-1$ is successful (and in the notation $\mathcal{U}(k_0,j)$ one is not counting j). □

We are finally ready to state the WCC bound for Algorithm 3.1 when the objective function is convex. To do that we combine Lemmas 4.3 and 4.4 and bound the number of successful iterations until the first unsuccessful one. By doing so we show below that direct search takes at most $\mathcal{O}(\epsilon^{1-\hat{p}})$ iterations after the first unsuccessful one to bring the norm of the gradient below $\epsilon \in (0,1)$.

Theorem 4.2 *Let Assumptions 4.2 hold. To reduce the norm of the gradient below $\epsilon \in (0,1)$, Algorithm 3.1 takes at most*

$$\kappa_3 \epsilon^{1-\hat{p}} + \kappa_4$$

iterations, where

$$\begin{aligned}
\kappa_3 &= 2(1-\log_{\beta_2}(\gamma))\frac{R}{\omega} - \log_{\beta_2}(\exp(1)),\\
\kappa_4 &= \log_{\beta_2}\left(\frac{\beta_1}{\alpha_{k_0}}\right) + \log_{\beta_2}\left(\frac{\beta_2}{\gamma}\omega_g^{\frac{1}{\min(1,p-1)}}\right) + \frac{f(x_0)-f_*}{\mathcal{C}\alpha_0^p},
\end{aligned}$$

and ω and ω_g are given in (7). When $p=2$, this number is of $\mathcal{O}(\nu^2\epsilon^{-1})$.

Proof. One can now use Lemmas 4.3 and 4.4

$$\begin{aligned}
\bar{k}-k_0 &= |\mathcal{S}(k_0,\bar{k})| + |\mathcal{U}(k_0,\bar{k})|\\
&\leq (1-\log_{\beta_2}(\gamma))|\mathcal{S}(k_0,\bar{k})| + \log_{\beta_2}(\beta_1/\alpha_{k_0})\\
&\quad + \frac{1}{\min(1,p-1)}\log_{\beta_2}(\omega_g\epsilon)\\
&\leq (1-\log_{\beta_2}(\gamma))\left(2\frac{R}{\omega}\epsilon^{1-\hat{p}}+1\right) + \log_{\beta_2}(\beta_1/\alpha_{k_0})\\
&\quad + \frac{1}{\min(1,p-1)}\log_{\beta_2}(\omega_g\epsilon).
\end{aligned}$$

From $1/\min(1,p-1) \leq \hat{p}-1$ (see the proof of Theorem 4.1) and the derivation (20) with $\bar{\epsilon}=\epsilon$,

$$\frac{1}{\min(1,p-1)}\log_{\beta_2}(\omega_g\epsilon) \;\leq\; (\hat{p}-1)\log_{\beta_2}(\omega_g\epsilon).$$

The proof is then completed by using $1-\log_{\beta_2}(\gamma) = \log_{\beta_2}(\beta_2/\gamma)$ and then by applying the bound (17) on k_0. □

To count the corresponding number of function evaluations we need first to factor out the dependence of n in the above bound. We know from [17] that, in this bound, only the minimum cosine measure of the positive spanning sets depends explicitly on n. One also knows from the positive spanning set formed by the coordinate vectors and their negatives that such minimum cosine measure can be set greater than or equal to $1/\sqrt{n}$, and thus $1/\omega \leq \mathcal{O}(n^{\frac{\hat{p}}{2}})$, where ω is given in (7). On the other hand, each poll step when using such positive spanning sets costs at most $\mathcal{O}(n)$ function evaluations. One then assumes, for compatibility with the cost of such poll steps, that the search step, when non-empty, takes at most $\mathcal{O}(n)$ function evaluations.

Corollary 4.1 *Let Assumptions 4.2 hold. Let* $\mathrm{cm}_{\min}$ *be at least a multiple of* $1/\sqrt{n}$ *and the number of function evaluations per iteration be at most a multiple of* n*. To reduce the norm of the gradient below* $\epsilon \in (0,1)$*, Algorithm 3.1 takes at most*

$$\mathcal{O}\left(n^{\frac{\hat{p}+2}{2}}\nu^{\hat{p}}\epsilon^{1-\hat{p}}\right)$$

function evaluations. When $p = 2$*, this number is of* $\mathcal{O}(n^2\nu^2\epsilon^{-1})$*.*

5 Global rate of direct search under strong convexity

A continuously differentiable function f is called strongly convex in $\mathbb{R}^n$ with constant $\mu > 0$ (notation $f \in \mathcal{F}^1_\mu(\mathbb{R}^n)$) if there exists a constant $\mu > 0$ such that, for any $x, y \in \mathbb{R}^n$,

$$f(x) \;\geq\; f(y) + \langle \nabla f(y), x - y\rangle + \frac{1}{2}\mu\|x - y\|^2.$$

As in [13, Equation (3.2)], by minimizing both sides of the above inequality in y for any $x \in \mathbb{R}^n$, strong convexity implies

$$f(x) - f_* \;\leq\; \frac{1}{2\mu}\|\nabla f(x)\|^2, \quad \forall x \in \mathbb{R}^n, \tag{25}$$

where $f_* = f(x_*)$ and x_* is the unique minimizer of f. Due to $\nabla f(x_*) = 0$ the first inequality above also implies

$$f(x) - \; f_* \;\geq\; \frac{1}{2}\mu\|x - x_*\|^2, \quad \forall x \in \mathbb{R}^n. \tag{26}$$

We will also make use of the following property (see [11, Equation (2.1.6)])

$$f(y) - f(x) - \langle \nabla f(x), y - x\rangle \;\leq\; \frac{\nu}{2}\|y - x\|^2 \tag{27}$$

for $f \in \mathcal{F}^1_{\nu,\mu}(\Omega)$ (meaning when f is strongly convex and ∇f is Lipschitz continuous with constant $\nu > 0$).

We are thus prepared to prove that the rate of convergence of function values and iterates for strongly convex functions is linear when $p = 2$. To avoid repeating the several assumptions in the statements of the results of this section we will combine them in the following one.

Assumption 5.1 *Consider the application of Algorithm 3.1 when* $\rho(t) = \mathcal{C}\,t^2$ *(*$p = 2$*),* $\mathcal{C} > 0$*, and* D_k *satisfies Assumption 3.1. Let* $f \in \mathcal{F}^1_{\nu,\mu}(\mathbb{R}^n)$*.*

As usual, we will start by considering first the case of the successful iterations.

Lemma 5.1 *Let Assumptions 5.1 hold. Let* k_0 *be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence* $\{x_k\}_{k\geq k_0}$ *such that*

$$f(x_k) - f_* \;<\; (1 - 2\omega\mu)^{|\mathcal{S}(k_0,k)|}(f(x_{k_0}) - f_*), \tag{28}$$

$$\|x_k - x_*\| \;<\; \sqrt{\frac{\nu}{\mu}}(1 - 2\mu\omega)^{\frac{1}{2}|\mathcal{S}(k_0,k)|}\|x_{k_0} - x_*\|, \tag{29}$$

where ω *is given in (7) and* $|\mathcal{S}(k_0,k)|$ *is the number of successful iterations between* k_0 *(including it) and* k*.*

Proof. Let j (with $k_0 < j \leq k$) be index of a successful iteration generated by Algorithm 3.1. Again, we can backtrack to nearest unsuccessful iteration k_ℓ (with $k_\ell \geq k_0$), and using the sufficient decrease condition, the step size updating rules, Corollary 3.1, and the definition of ω in (7), we obtain

$$\begin{aligned} f(x_j) - f(x_{j+1}) &> \mathcal{C}\alpha_j^2 \\ &\geq \mathcal{C}\beta_1^2\alpha_{k_\ell}^2 \\ &\geq \omega\|\nabla f(x_{k_\ell})\|^2 \\ &\geq 2\omega\mu(f(x_{k_\ell}) - f_*) \\ &> 2\omega\mu(f(k_\ell) - f_*), \end{aligned}$$

where the fourth inequality follows from inequality (25). Hence,

$$f(x_{j+1}) - f_* \ < \ (1 - 2\omega\mu)(f(x_j) - f_*).$$

A repeatedly application of the above inequality will lead us to (28).

Now, the application of inequalities (26), (28), and (27) with $y = x_{k_0}$ and $x = x_*$ gives us

$$\begin{aligned} \frac{\mu}{2}\|x_k - x_*\|^2 &\leq f(x_k) - f_* \\ &< (1 - 2\mu\omega)^{|\mathcal{S}(k_0,k)|}(f(x_{k_0}) - f_*) \\ &\leq (1 - 2\mu\omega)^{|\mathcal{S}(k_0,k)|}\frac{\nu}{2}\|x_{k_0} - x_*\|^2, \end{aligned}$$

yielding (29). □

Now one needs to take care of the number of unsuccessful iterations. The assumption of strongly convexity will lead to a bound better than (13).

Lemma 5.2 *Let Assumptions 5.1 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence $\{x_k\}_{k\geq k_0}$ such that*

$$|\mathcal{U}(k_0,k)| \ \leq \ \left\lceil \omega_1|\mathcal{S}(k_0,k)| + \omega_2 + \log_{\beta_2}(\omega_g\sqrt{2\mu(f(x_k) - f_*)}) \right\rceil, \tag{30}$$

$$|\mathcal{U}(k_0,k)| \ \leq \ \left\lceil \omega_1|\mathcal{S}(k_0,k)| + \omega_2 + \log_{\beta_2}(\mu\omega_g\|x_k - x_*\|) \right\rceil, \tag{31}$$

where ω_g is given in (7), ω_1 and ω_2 are given in (14), and $|\mathcal{S}(k_0,k)|$ and $|\mathcal{U}(k_0,k)|$ are the number of successful and unsuccessful iterations between k_0 (including it) and k, respectively.

Proof. From inequality (25), one has for each unsuccessful iteration k_i (with $k_0 \leq k_i \leq k$)

$$\|\nabla f(x_{k_i})\|^2 \ \geq \ 2\mu(f(x_{k_i}) - f_*) \ \geq \ 2\mu(f(x_k) - f_*).$$

Now, by an argument like in the proof of the Lemma 4.2, but using $\sqrt{2\mu(f(x_k) - f_*)}$ instead of $(f(x_k) - f_*)/R$, one obtains

$$f(x_k) - f_* \ \leq \ \frac{1}{2\mu\omega_g^2}\left(\frac{\alpha_{k_0}}{\beta_1}\gamma^{|\mathcal{S}(k_0,k)|}\beta_2^{|\mathcal{U}(k_0,k)|}\right)^2.$$

In turn, this inequality and (26) imply

$$\|x_k - x_*\| \;\leq\; \frac{1}{\mu\omega_g}\left(\frac{\alpha_{k_0}}{\beta_1}\gamma^{|\mathcal{S}(k_0,k)|}\beta_2^{|\mathcal{U}(k_0,k)|}\right),$$

and the proof can be finished by applying $\log_{\beta_2}$ and noting that $\beta_2 < 1$ and $\omega_1 \geq 0$. $\square$

Lemmas 5.1 and 5.2 result in a linear convergence rate (when $p = 2$) for the absolute error in function values and iterates after the first unsuccessful iteration.

Theorem 5.1 *Let Assumptions 5.1 hold. Let k_0 be the index of the first unsuccessful iteration (which must exist from Lemma 3.1). Then Algorithm 3.1 generates a sequence $\{x_k\}_{k\geq k_0}$ such that*

$$f(x_k) - f_* \;<\; \beta^{\frac{1}{\kappa_5}(k-\kappa_6)},$$
$$\|x_k - x_*\| \;<\; \beta^{\frac{1}{2\kappa_5}(k-\kappa_7)},$$

where

$$\begin{aligned}
\kappa_5 &= (1+\omega_1)\log_{1-2\omega\mu}(\beta) + \log_{\beta_2}(\sqrt{\beta}), \quad \beta = \min(\beta_2, 1-2\mu\omega),\\
\kappa_6 &= \frac{f(x_0)-f_*}{\mathcal{C}\alpha_0^2} - (1+\omega_1)\log_{1-2\omega\mu}(f(x_{k_0})-f_*) + \omega_2 + \log_{\beta_2}(\omega_g\sqrt{2\mu}),\\
\kappa_7 &= \frac{\nu\|x_0-x_*\|^2}{2\mathcal{C}\alpha_0^2} - (1+\omega_1)\log_{1-2\omega\mu}\left(\frac{\nu}{\mu}\|x_{k_0}-x_*\|^2\right) + \omega_2 + \log_{\beta_2}(\omega_g\mu),
\end{aligned}$$

ω and ω_g are given in (7), and ω_1 and ω_2 are given in (14).

Proof. From inequalities (28) and (30) one has

$$\begin{aligned}
k - k_0 &= |\mathcal{U}(k_0,k)| + |\mathcal{S}(k_0,k)|\\
&\leq (1+\omega_1)|\mathcal{S}(k_0,k)| + \omega_2 + \log_{\beta_2}\left(\omega_g\sqrt{2\mu(f(x_k)-f_*)}\right)\\
&< (1+\omega_1)\log_{1-2\omega\mu}\left(\frac{f(x_k)-f_*}{f(x_{k_0})-f_*}\right) + \omega_2 + \log_{\beta_2}\left(\omega_g\sqrt{2\mu(f(x_k)-f_*)}\right)\\
&= \left[(1+\omega_1)\log_{1-2\omega\mu}(\beta) + \log_{\beta_2}(\sqrt{\beta})\right]\log_\beta(f(x_k)-f_*)\\
&\quad - (1+\omega_1)\log_{1-2\omega\mu}(f(x_{k_0})-f_*) + \omega_2 + \log_{\beta_2}(\omega_g\sqrt{2\mu}).
\end{aligned}$$

Then, from inequality (17), one obtains

$$\begin{aligned}
k &- \left[\frac{f(x_0)-f_*}{\mathcal{C}\alpha_0^2} - (1+\omega_1)\log_{1-2\omega\mu}(f(x_{k_0})-f_*) + \omega_2 + \log_{\beta_2}(\omega_g\sqrt{2\mu})\right]\\
&< \left[(1+\omega_1)\log_{1-2\omega\mu}(\beta) + \log_{\beta_2}(\sqrt{\beta})\right]\log_\beta(f(x_k)-f_*)
\end{aligned}$$

which proves the first part of the theorem.

By similar arguments, but using now inequalities (29) and (31), it results that

$$\begin{aligned} k - k_0 &= |\mathcal{U}(k_0,k)| + |\mathcal{S}(k_0,k)| \\ &\leq (1+\omega_1)|\mathcal{S}(k_0,k)| + \omega_2 + \log_{\beta_2}(\omega_g\mu\|x_k - x_*\|) \\ &< (1+\omega_1)\log_{1-2\omega\mu}\left(\frac{\mu}{\nu}\frac{\|x_k-x_*\|^2}{\|x_{k_0}-x_*\|^2}\right) + \omega_2 + \log_{\beta_2}(\omega_g\mu\|x_k - x_*\|) \\ &= \left[2(1+\omega_1)\log_{1-2\omega\mu}(\beta) + \log_{\beta_2}(\beta)\right]\log_\beta(\|x_k - x_*\|) \\ &\quad - (1+\omega_1)\log_{1-2\omega\mu}\left(\frac{\nu}{\mu}\|x_{k_0}-x_*\|^2\right) + \omega_2 + \log_{\beta_2}(\omega_g\mu). \end{aligned}$$

Again using inequality (17), but now followed by (27) with $x = x_0$ and $y = x_*$, yields

$$\begin{aligned} k \;-\; &\left[\frac{\nu\|x_0-x_*\|^2}{2\mathcal{C}\alpha_0^2} - (1+\omega_1)\log_{1-2\omega\mu}\left(\frac{\nu}{\mu}\|x_{k_0}-x_*\|^2\right) + \omega_2 + \log_{\beta_2}(\omega_g\mu)\right] \\ < \;&\left[2(1+\omega_1)\log_{1-2\omega\mu}(\beta) + \log_{\beta_2}(\beta)\right]\log_\beta(\|x_k - x_*\|), \end{aligned}$$

which proves the second part. □

The result of Theorem 5.1 improves significantly what has been known for direct search. In fact, it was proved in [4] that the absolute error for unsuccessful iterates exhibits an r-linear rate of convergence under the following assumptions: α_k is monotonically nonincreasing and x_k is sufficiently close to a point x_* for which $\nabla f(x_*) = 0$ and $\nabla^2 f(x)$ is positive definite around x_*. Our result is therefore stronger since (i) the r-linear rate is over the all sequence $\|x_k - x_*\|$, whether k is successful or not, (ii) only first order continuously differentiability is assumed, and (iii) α_k can be increased at successful iterations.

6 A comparison with random Gaussian methods

In this section we will report the outcome of numerical experiments involving direct search (Algorithm 3.1) and the random Gaussian methods of Nesterov [12] and Duchi et al. [5] for the unconstrained minimization (1) of a smooth, convex function f. We start by describing the simplest version of the method in [12] when f is smooth. In the scheme below, $u_k \in \mathbb{R}^n$ is drawn from a Gaussian distribution with correlation operator B^{-1}.

Algorithm 6.1 (Random Gaussian method)

Initialization

Choose x_0.

For $k = 0, 1, 2, \ldots$

1. Generate u_k and compute $g_\lambda(x_k) = \frac{f(x_k+\lambda u_k)-f(x_k)}{\lambda} B u_k$.
2. Choose a step size $h_k > 0$. Compute $x_{k+1} = x_k - h_k B^{-1} g_\lambda(x_k)$.

n	accuracy							
	2.0e-3	9.8e-4	4.9e-4	2.4e-4	1.2e-4	6.1e-5	3.1e-5	1.5e-5
2	351	402	451	502	551	599	648	699
4	2738	3169	3588	4019	4439	4848	5256	5694
8	24627	29033	33313	37724	42005	46184	50364	54841
16	251313	304464	356115	409297	460939	511327	561680	615513

Table 1: Average performance of the random Gaussian method (Algorithm 6.1) for the minimization of (34).

One can see that $[f(x_k+\lambda u_k)-f(x_k)]/\lambda$ is a finite-difference approximation to the derivative of $f(x_k+\alpha u_k)$ with respect to α. In [12] it is also considered a central-difference approximation to this derivative, but we do not present it here as it performed worse. It is proved in [12] that Algorithm 6.1 achieves a sublinear rate of $1/k$ in function values for $f \in \mathcal{F}^1_\nu(\mathbb{R}^n)$, $X^f_* \neq \emptyset$, in the case where λ is chosen sufficiently small

$$\lambda \;\leq\; \lambda_{\max} \;=\; \frac{5}{3(n+4)}\sqrt{\frac{\epsilon}{2\nu}} \tag{32}$$

and h_k is set constantly to

$$h_k \;=\; \frac{1}{4(n+4)\nu}. \tag{33}$$

The convergence rate in function values is shown to be r-linear when f is strongly convex.

In the experiments reported in this paper to test Algorithms 3.1 and 6.1, we used the same positive definite quadratic as in [12],

$$f(x) \;=\; \frac{1}{2}x_1^2 + \frac{1}{2}\sum_{i=1}^{n-1}(x_{i+1}-x_i)^2 + \frac{1}{2}x_n^2 - x_1. \tag{34}$$

Algorithm 6.1 was tested, as in [12], with the constant step size (33) and the maximum allowed value λ_{max} in (32) where $\epsilon = 2^{-16}$.

Algorithm 3.1 was run using $D_\oplus = [I\ -I]$ as the set of poll directions. The step size α_k was kept unchanged after a successful iteration, and contracted by a factor of $1/2$ after an unsuccessful one. The forcing function $\rho(t)$ was $10^{-3}t^2$. Regarding the order of the function evaluations in the poll step, several strategies were tried with relatively equivalent results, but we report here results with the random polling strategy, where the poll directions are ordered randomly before the start of each iteration.

The results are reported in Table 1 for Algorithm 6.1 and in Table 2 for Algorithm 3.1. The tables display the number of function evaluations needed to reach various optimal value accuracies. All the figures reported there are averages of 20 runs made for the fixed starting point $x_0 = 0$. As in [12], the accuracy levels are $2^{-(r+7)}$, $r = 2, \ldots, 9$. As one can see from the tables, random Gaussian performs much worse than direct search. Setting instead $h_k = 1/(k+1)$, $k = 0, 1, \ldots$ in Step 2 of Algorithm 6.1 did not seem to improve the performance.

One modification that does improve the numerical performance of random Gaussian significantly is to randomize the point at which the finite-difference approximation is taken (which

n	accuracy							
	2.0e-3	9.8e-4	4.9e-4	2.4e-4	1.2e-4	6.1e-5	3.1e-5	1.5e-5
2	26	34	34	39	39	45	45	50
4	136	146	168	187	207	226	247	263
8	855	970	1061	1202	1338	1480	1585	1716
16	5146	6175	7216	8296	9559	10568	11543	12613

Table 2: Average performance of direct search (Algorithm 3.1) for the minimization of (34).

then naturally leads to two function evaluations per iteration). The approach in [5] is designed for stochastic optimization but one can consider its counterpart tailored for deterministic optimization in the context of Derivative-Free Optimization (DFO):

Algorithm 6.2 (Two-points random Gaussian method)

Initialization

Choose x_0 and two non-increasing sequences of positive finite-difference steps $\{\lambda_{1,k}\}_{k=0}^{\infty}$, $\{\lambda_{2,k}\}_{k=0}^{\infty}$.

For $k = 0, 1, 2, \ldots$

1. Generate $u_{1,k}$ and $u_{2,k}$ (uniformly and independently) and compute

$$g_{\lambda_{1,k},\lambda_{2,k}}(x_k) \;=\; \frac{f(x_k + \lambda_{1,k}u_{1,k} + \lambda_{2,k}u_{2,k}) - f(x_k + \lambda_{1,k}u_{1,k})}{\lambda_{2,k}} u_{2,k}.$$

2. Choose a step size $h_k > 0$. Compute $x_{k+1} = x_k - h_k g_{\lambda_{1,k},\lambda_{2,k}}(x_k)$.

Following [5, Theorem 2], one can select the finite-difference steps as

$$\lambda_{1,k} \;=\; \lambda_{\max}\frac{r_0}{k+1} \quad \text{and} \quad \lambda_{2,k} \;=\; \lambda_{\max}\frac{r_0}{n^2(k+1)^2}, \quad k = 0, 1, 2, \ldots \tag{35}$$

where λ_{max} is given by (32), with $\epsilon = 2^{-16}$, and $r_0 = \text{dist}(x_0, X_*^f)$, with X_*^f the set of minimizers of f. The step size can be set to $h = 1/\nu$, where ν is the Lipschitz constant of gradient of f. These choices are, of course, unrealistic in DFO, or even in Nonlinear Programming, since neither Lipschitz constants can be reasonably approximated nor the distance to the solutions set can be computed.

We tested Algorithm 6.2 on (34), taking also averages of 20 runs from the fixed starting point $x_0 = 0$. The results are presented in Table 3 in the same way as in Tables 1–2. We observe that the two-points random Gaussian method (Algorithm 6.2) performs noticeably better than the random Gaussian one (Algorithm 6.1) but still worse than direct search (Algorithm 3.1).

We also tested Algorithms 3.1 and 6.2 on five unconstrained problems from the CUTEr collection [8], with objective function convex and dimension $n = 8$. The numerical results for these problems are presented in Tables 4–5, organized as the previous ones. The numbers of function evaluations are averages of 20 runs departing always from the initial point given by

n	accuracy							
	2.0e-3	9.8e-4	4.9e-4	2.4e-4	1.2e-4	6.1e-5	3.1e-5	1.5e-5
2	27	32	35	39	43	47	49	54
4	168	193	216	242	265	290	314	340
8	1039	1222	1406	1593	1775	1952	2129	2322
16	6284	7596	8868	10143	11360	12574	14067	16334

Table 3: Average performance of the two-points random Gaussian method (Algorithm 6.2) for the minimization of (34).

Problem	accuracy							
	2.0e-3	9.8e-4	4.9e-4	2.4e-4	1.2e-4	6.1e-5	3.1e-5	1.5e-5
`arglinc`	179	186	192	199	205	211	217	224
`dqrtic`	78634	79256	80111	81393	83148	85567	88997	94211
`vardim`	23409	24582	25738	26922	28082	29225	30378	31590
`nondquar`	2401	4138	6988	11825	19219	29475	43985	66658
`powellsg`	11487	14659	19173	27612	43104	66514	100775	154520

Table 4: Average performance of the two-points random Gaussian method (Algorithm 6.2) on five CUTEr problems (convex and unconstrained).

CUTEr. One sees clearly from these results that direct search (Algorithm 3.1) performs overall much better than two-points random Gaussian (Algorithm 6.2). The stopping criterion used in both Algorithms 6.1 and 6.2 considers the value of f at x_{k+1}. We also tried to use the minimum among this value and all the values of f calculated during the k-th iteration but the impact of such a change is negligible (in any of the problems reported in this section).

It is important to remark, though, that a huge advantage has been given to Algorithm 6.2 by using the distance r_0 to the solutions set and the value ν of the Lipschitz constant of the gradient of f. In fact, in all these problems, we used the exact value of r_0 in Algorithm 6.2, see (35). Further, the value of ν (used in the step size h_k update (33) for Algorithm 6.2) was set exactly for problem `arglinc` and approximated around the minimizer for problems `dqrtic`, `vardim`, `nondquar`, and `powellsg`. Still, direct search performed significantly better. This does not come as a surprise since both Algorithms 6.1 and 6.2 do not accept steps and do not adjust step sizes depending on some form of decrease as does direct search.

7 Conclusions

To our knowledge it is the second time that a derivative-free method is shown to exhibit a worst case complexity (WCC) bound of $\mathcal{O}(\epsilon^{-1})$ in the convex case, following the random Gaussian approach [12], but the first time for a deterministic approach (where the bound is not taken in expectation). In fact we have proved that a maximum of $\mathcal{O}(\epsilon^{-1})$ iterations and $\mathcal{O}(n^2\epsilon^{-1})$ function evaluations are required to compute a point for which the norm of the gradient of the

Problem	accuracy							
	2.0e-3	9.8e-4	4.9e-4	2.4e-4	1.2e-4	6.1e-5	3.1e-5	1.5e-5
arglinc	195	235	235	235	235	235	235	235
dqrtic	69	69	69	69	69	69	69	69
vardim	2796	2996	3198	3478	3716	3935	4156	4460
nondquar	180	211	261	330	800	2196	4668	7905
powellsg	1223	1554	2004	2685	3601	5107	6713	9245

Table 5: Average performance of the direct search (Algorithm 3.1) on five CUTEr problems (convex and unconstrained).

.

objective function f is smaller than $\epsilon \in (0,1)$ (see Theorem 4.2 and Corollary 4.1 when $p = 2$ in the forcing function). In addition we proved that the absolute error $f(x_k) - f_*$ decreases at a sublinear rate of $1/k$ (see Theorem 4.1). Such results are global in the sense of not depending on the proximity of the initial iterate to the solutions set.

These WCC bounds and global rates were proved when the solutions set is bounded or, when that is not the case, when the supreme distance from any point in the initial level set to the solutions set is bounded (Assumption 4.1). A particular case is strong convexity where the solution set is a singleton. In such a case, we went a step further (when $p = 2$ in the forcing function) and showed that $f(x_k) - f_*$ decreases r-linearly and so does $\|x_k - x_*\|$ (see Theorem 5.1).

There are some rare pathological instances where Assumption 4.1 does not hold. An example is the following two-dimensional convex function

$$f(x,y) \;=\; \sqrt{x^2+y^2} - x.$$

The minimum of f is equal to zero and the solutions set is $X_*^f = \{(x,0) : x \geq 0\}$. Let $\varsigma = f(x_0,y_0) > 0$ be given for some (x_0,y_0). Then

$$f^{-1}(\{\varsigma\}) \;=\; \{z \in \mathbb{R}^2 : f(z) = \varsigma\} \;=\; \{(t^2-\varsigma^2)/2\varsigma, t)\}_{t\in\mathbb{R}}.$$

Thus, for $z = ((t^2-\varsigma^2)/2\varsigma, t) \in f^{-1}(\{\varsigma\})$, one has $\mathrm{dist}(z, X_*^f) \geq |t|$, which implies

$$\sup_{z\in L_f(x_0,y_0)} \mathrm{dist}(z, X_*^f) \;\geq\; \sup_{z\in f^{-1}(\{\varsigma\})} \mathrm{dist}(z, X_*^f) \;\geq\; \sup_{t\in\mathbb{R}} |t| \;=\; +\infty.$$

Notice that this function is not continuously differentiable at the origin but an alternative, smoothed version could be instead considered.

Acknowledgments

We would like to thank one anonymous referee for the numerous insightful observations which led to a much improved version of the paper. We thank also Zaikun Zhang for several interesting discussions.

References

[1] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization. *SIAM J. Optim.*, 20:2833–2852, 2010.

[2] C. Cartis, N. I. M. Gould, and Ph. L. Toint. On the oracle complexity of first-order and derivative-free algorithms for smooth nonconvex minimization. *SIAM J. Optim.*, 22:66–86, 2012.

[3] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2009.

[4] E. D. Dolan, R. M. Lewis, and V. Torczon. On the local convergence of pattern search. *SIAM J. Optim.*, 14:567–583, 2003.

[5] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono. Optimal rates for zero-order convex optimization: the power of two function evaluations. arXiv:1312.2139v2, 2014.

[6] R. Garmanjani and L. N. Vicente. Smoothing and worst-case complexity for direct-search methods in nonsmooth optimization. *IMA J. Numer. Anal.*, 33:1008–1028, 2013.

[7] C. Cartis N. I. M. Gould and Ph. L. Toint. Adaptive cubic regularisation methods for unconstrained optimization. Part II: Worst-case function- and derivative-evaluation complexity. *Math. Program.*, 130:295–319, 2011.

[8] N. I. M. Gould, D. Orban, and P. L. Toint. CUTEr, a Constrained and Unconstrained Testing Environment, revisited. *ACM Trans. Math. Software*, 29:373–394, 2003.

[9] S. Gratton, A. Sartenaer, and Ph. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization. *SIAM J. Optim.*, 19:414–444, 2008.

[10] T. G. Kolda, R. M. Lewis, and V. Torczon. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Rev.*, 45:385–482, 2003.

[11] Y. Nesterov. *Introductory Lectures on Convex Optimization.* Kluwer Academic Publishers, Dordrecht, 2004.

[12] Y. Nesterov. Random gradient-free minimization of convex functions. Technical Report 2011/1, CORE, 2011.

[13] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM J. Optim.*, 22:341–362, 2012.

[14] Y. Nesterov and B. T. Polyak. Cubic regularization of Newton's method and its global performance. *Math. Program.*, 108:177–205, 2006.

[15] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer-Verlag, Berlin, second edition, 2006.

[16] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, 1970.

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