A second-order globally convergent direct-search method and its worst-case complexity

S. Gratton[*] C. W. Royer[†] L. N. Vicente[‡]

June 4, 2015

Abstract

Direct-search algorithms form one of the main classes of algorithms for smooth unconstrained derivative-free optimization, due to their simplicity and their well-established convergence results. They proceed by iteratively looking for improvement along some vectors or directions. In the presence of smoothness, first-order global convergence comes from the ability of the vectors to approximate the steepest descent direction, which can be quantified by a first-order criticality (cosine) measure. The use of a set of vectors with a positive cosine measure together with the imposition of a sufficient decrease condition to accept new iterates leads to a convergence result as well as a worst-case complexity bound.

In this paper, we present a second-order study of a general class of direct-search methods. We start by proving a weak second-order convergence result related to a criticality measure defined along the directions used throughout the iterations. Extensions of this result to obtain a true second-order optimality one are discussed, one possibility being a method using approximate Hessian eigenvectors as directions (which is proved to be truly second-order globally convergent). Numerically guaranteeing such a convergence can be rather expensive to ensure, as it is indicated by the worst-case complexity analysis provided in this paper, but turns out to be appropriate for some pathological examples.

1 Introduction

Derivative-free algorithms for unconstrained optimization attempt to minimize a function $f : \mathbb{R}^n \to \mathbb{R}$ without using any form of derivatives. Such methods are particularly relevant when the function f comes from a simulation code; indeed, the user seldom has access to the source code itself, but rather to the outcome of computer runs, preventing the use of automatic differentiation techniques to approximate the derivatives. In addition, such simulations are often characterized by the particularly high cost of a single evaluation, which tends to make the use of

[*] ENSEEIHT, INPT, rue Charles Camichel, B.P. 7122 31071, Toulouse Cedex 7, France (serge.gratton@enseeiht.fr).

[†] IRIT-ENSEEIHT, rue Charles Camichel, B.P. 7122 31071, Toulouse Cedex 7, France (clement.royer@enseeiht.fr.) Support for this author comes from a doctoral grant of Université Toulouse III Paul Sabatier.

[‡] CMUC, Department of Mathematics, University of Coimbra, 3001-501 Coimbra, Portugal (lnv@mat.uc.pt). Support for this research was provided by FCT under grant UID/MAT/00324/2013 and by the Réseau Thématique de Recherche Avancée, Fondation de Coopération Sciences et Technologies pour l'Aéronautique et l'Espace, under the grant FILAOS.

finite-differences approximations too expensive. As a result, derivative-based methods are not suited for such problems, for which derivative-free optimization (DFO) algorithms have been developed. For more general information on DFO and particular examples of applications, we refer the reader to the monograph by Conn, Scheinberg, and Vicente [13].

Among the most popular derivative-free methods are direct-search algorithms of directional type that rely on exploring the variable space through directions specified within the algorithm. At each iteration, a polling set of directions is used to determine new points at which the function is evaluated; if such exploratory moves manage to find a point that reduces the function value (possibly minus a term determined by the algorithm to make the decrease sufficient enough), then this point becomes the current iterate. Typical instances of direct search include Pattern Search [35], Generating Set Search [28], and Mesh Adaptive Direct Search [4]. Under appropriate smoothness and boundedness assumptions on f, these frameworks exhibit first-order convergence provided the directions satisfy suitable properties. The usual requirement on the polling sets is that they contain Positive Spanning Sets (PSSs), which are known to generate the variable space by positive linear combination [15]. In a smooth setting, the use of those sets at each iteration ensures that one of the directions always makes an acute angle with the steepest descent one. This guarantees, eventually, that a satisfying function decrease is produced and, as a result, that a subsequence of iterates converges towards a first-order stationary point (such convergence is called global since it is independent of the starting point).

Second-order global convergence is usually less explored in direct search, even though several algorithms attempt to use second-order aspects in their framework [7, 14, 31]. In fact, proving second-order results for such zeroth-order methods necessitates to strengthen the aforementioned descent requirements, thus raising the question of their practical relevance. Still, second-order results for classical direct-search schemes have been established [1, 2, 3]. In all of those proofs, the concept of a refining direction is used for proving second-order convergence. Given a subsequence of iterations, one defines the set of refining directions with respect to this subsequence as the closure of the limit points of the corresponding polling set subsequence [4]. One may thus consider a subsequence of iterates converging to a first-order critical point and the associated set of refining directions in order to state a second-order-type result. In the case of Pattern Search, the second-order optimality cannot be proved with respect to all vectors in the space or, equivalently, all directions in the unit sphere. However, if a given orthonormal basis and its opposite appear infinitely many in the subsequence, it is possible to prove that the limit point satisfies a *pseudo-second order optimality* property which is that the Hessian is positive semidefinite in the directions of the basis [1, Theorem 3.9]. Regarding MADS-type methods, it was proved that if the set of refining directions corresponding to a first-order convergent subsequence of iterates is dense in the unit sphere, then the limit point of the subsequence satisfies second-order necessary optimality conditions [2, Theorem 4.8]. The result is the same for the Generating Set Search method with Curvature Information, which uses a Hessian approximation to gather second-order information. In fact, the quality of this approximation is crucial for second-order optimality [3, Theorem 3.6] and such an assumption is close to what is assumed in second-order convergent model-based methods [12].

The aforementioned results are obtained considering the limits of first-order convergent subsequences of iterations and the properties at these limits, and this reasoning is inspired by the first-order one [4]. We are interested in an other proof scheme that reasons at the iteration level. Indeed, it is known that one may relate the step size to a certain measure of first-order optimality (defined by the largest cosine of the angle made by the directions in the polling sets with the negative gradient). Although, in the general case, this measure is weaker than the gradient norm, it is possible to arrive at a first-order result when using polling sets that keep such a measure bounded away from zero [28]. One might thus wonder if it is possible to study second-order properties also at the iteration level.

It is then the first goal of this paper to define an appropriate measure of second-order criticality. As opposed to the first-order case, this measure is not expected to lead to classic second-order results in general. However, having defined this second-order measure, we examine the conditions under which it enables to prove second-order global convergence. We know that such a convergence is achieved by some direct-search instances relying either on a sequence of directions dense in the unit sphere or on a Hessian approximation. Even though those requirements have a practical cost, they make sense considering what we ask from our method. The second goal of this paper is to develop a direct-search scheme ultimately relying on a Hessian approximation for which we prove global convergence to second-order critical points using our second-order measure.

Finally, we aim to provide a bound on the number of function evaluations needed to reach an approximate satisfaction of a second-order criterion. Note that this is only possible if we are able to measure the progress made towards optimality at each iteration, which is indeed what we hope to achieve using a second-order criticality measure. Such worst-case complexity matters have already been investigated in direct search for first-order optimality [36, 16, 17, 22] (see also [29] for the case of no step size increase and [25] for the probabilistic case). On the other hand, second-order worst-case complexity bounds have been derived in unconstrained smooth non-convex optimization [9, 10] for derivative-based methods. It is thus natural to look for similar results for direct search, in a second-order context.

The remaining of this paper is organized as follows. In Section 2, we present the framework of our analysis and its first-order behavior, and study the second-order properties that can be proved for such a class of methods. Section 3 discusses possible ways to extend the weak second-order results to obtain second-order convergence in the usual sense; as an illustration, we introduce a second-order globally convergent algorithm. This latter method is again considered in Section 4 for the derivation of a worst case complexity result. Preliminary numerical experiments are presented in Section 5 to evaluate the appropriateness and practical cost of the second-order requirements. Throughout the paper, the notation $\mathcal{O}(A)$ will stand for a scalar times A, with this scalar depending solely on the problem considered or constants from the algorithm. When appropriate, the dependence on the problem dimension will explicitly appear in A.

2 Weak second-order results for direct search based on sufficient decrease

In this section, we present an elementary direct-search algorithm in which new points are considered if they satisfy a *sufficient decrease*. This is the baseline of many algorithms in the Generating Set Search class [28]. We start by recalling the first-order convergence results that can be certified when one uses a particular kind of polling sets, then show that for any of those, there is also a weak form of second-order optimality that can be proved.

2.1 A general direct-search framework

We introduce below the framework that we will use in our study. In the rest of the document, it will be referred to as BDS, for Basic Direct Search.

Algorithm 2.1 (Basic Direct-Search)

Init. *Choose* $x_0 \in \mathbb{R}^n,\ \alpha_{\max} > \alpha_0 > 0,\ 0 < \theta < 1 < \gamma$, *and a* forcing function $\rho : \mathbb{R}^+ \to \mathbb{R}^+$.

1. *Set* $k = 0$.

2. *Generate a polling set* D_k.

3. If *there exists* $d_k \in D_k$ *such that*

$$f(x_k + \alpha_k\, d_k) \ < \ f(x_k) - \rho(\alpha_k), \tag{2.1}$$

then *declare iteration* k successful, *set* $x_{k+1} = x_k + \alpha_k\, d_k$, $\alpha_{k+1} = \min\{\gamma\,\alpha_k, \alpha_{\max}\}$, $k = k+1$ *and go to 2.*

4. Otherwise *declare the iteration* unsuccessful, *set* $x_{k+1} = x_k, \alpha_{k+1} = \theta\,\alpha_k, k = k+1$ *and go to 2.*

We refer to α_k as the step size relative to the k-th iteration and to x_k as the k-th iterate. The set of successful and unsuccessful iterations will be denoted by S and U, respectively. Note that one can add an optional search step before polling along the directions of D_k; we omit this step here since it does not affect the convergence nor the complexity analysis.

Under first-order assumptions on the objective function, one can analyse both the first-order convergence [28] and the worst-case complexity [36] of the algorithm. In our case, however, we will extend these assumptions beyond what is needed for first order, since we are interested in second-order results. We start by the assumptions on the objective function f.

Assumption 2.1 *The function f is twice continuously differentiable with Lipschitz-continuous gradient and Hessian, of respective Lipschitz constants ν_g and ν_H.*

Assumption 2.2 *f is bounded from below on $\mathcal{L}(x_0) = \{x \in \mathbb{R}^n : f(x) \le f(x_0)\}$ and we denote by f_{low} a lower bound.*

The next assumption concerns the forcing function ρ.

Assumption 2.3 *The forcing function $\rho : \mathbb{R}^+ \to \mathbb{R}^+$ satisfies the following properties:*

i) *ρ is non-decreasing;*

ii) *$\rho(\alpha) = o(\alpha^2)$ when $\alpha \to 0^+$.*

Note that when $t \to 0^+$, one has $\rho(\alpha) = o(\alpha)$, as is required for proving first-order convergence.

We finally look at the suitable properties for the direction sets, that involve a particular class of sets called Positive Spanning Sets (PSS). The use of PSS, that span $\mathbb{R}^n$ by positive linear combinations, is a common way of ensuring that a direct-search method considers descent directions. Indeed, since f is continuously differentiable, a PSS always contains a direction that makes an acute angle with the negative gradient, thus ensuring descent if the step size is sufficiently small [28]. Global convergence in then obtained by assuming that there exists $\kappa_g > 0$ such that:

$$\forall k, \ \operatorname{cm}(D_k) = \min_{v \in \mathbb{R}^n \setminus \{0\}} \max_{d \in D_k} \frac{d^\top v}{\|d\| \, \|v\|} \ \geq \ \kappa_g. \tag{2.2}$$

In other words, the sequence of *cosine measures* $\{\operatorname{cm}(D_k)\}_k$ is bounded away from zero; this represents our first requirement on the directions sets.

Existing second-order convergence analyses are based on *symmetric* polling sets for a refining subsequence. Such symmetric sets satisfy the property $-D = \{-d \mid d \in D\} = D$. Their use allows to take advantage of evaluations of the function in opposite directions, which is fundamental if one aims to consider the second-order difference scheme

$$f(x_k + \alpha_k \, d_k) - 2\, f(x_k) + f(x_k - \alpha_k \, d_k) \ = \ \alpha_k^2 \, d_k^\top \, \nabla^2 f(x_k) \, d_k + \mathcal{O}(\alpha_k^3) \tag{2.3}$$

that serves as an approximation of the second-order directional derivative. This being said, one can still prove results in the more general case where each polling set admits a symmetric subset. This case is described in the following assumption.

Assumption 2.4 *The polling sets D_k are finite Positive Spanning Sets of unitary vectors, such that the* symmetric part of D_k *defined by*

$$V_k \ = \ \{d \in D_k \mid -d \in D_k\}$$

is not empty, and that the sequence of cosine measures is bounded from below by $\kappa_g > 0$.

A relevant example of such a sequence is $D_k = [Q \ -Q]$, where Q is a rotation matrix; in that case, we have $D_k = V_k$ at each iteration. Lemma 2.1 enlightens the second-order property that can be stated using V_k.

Lemma 2.1 *Under Assumptions 2.1 and 2.4, consider an iteration of index $k \in U$. One has*

$$\|\nabla f(x_k)\| \ \leq \ \kappa_g^{-1} \left(\frac{\rho(\alpha_k)}{\alpha_k} + \frac{\nu_g}{2} \, \alpha_k \right) \tag{2.4}$$

and

$$\min_{d \in V_k} d^\top \, \nabla^2 f(x_k) \, d \ \geq \ - \left(\frac{2\, \rho(\alpha_k)}{\alpha_k^2} + \frac{\nu_H}{3} \, \alpha_k \right). \tag{2.5}$$

Proof. We begin by proving (2.4). The proof is taken from [28] and is now well known. If $\nabla f(x_k) = 0$, the result trivially holds, thus we assume in what follows that $\nabla f(x_k) \neq 0$. Since the iteration is unsuccessful, for each $d \in D_k$, one has

$$-\rho(\alpha_k) \ \leq \ f(x_k + \alpha_k \, d) - f(x_k). \tag{2.6}$$

Thus, a first-order Taylor expansion of $f(x_k + \alpha_k \, d)$ together with Assumption 2.1 leads to

$$-\rho(\alpha_k) \ \leq \ \alpha_k \, \nabla f(x_k)^\top \, d + \alpha_k \int_0^1 \left[\nabla f(x_k + t\alpha_k \, d) - \nabla f(x_k) \right]^\top \, d \, dt \ \leq \ \alpha_k \, \nabla f(x_k)^\top \, d + \frac{\nu_g}{2} \, \alpha_k^2. \tag{2.7}$$

Hence

$$-d^\top \nabla f(x_k) \;\leq\; \frac{\rho(\alpha_k)}{\alpha_k} + \frac{\nu_g}{2}\alpha_k. \tag{2.8}$$

Due to Assumption 2.4, we know that

$$\max_{d \in D_k} -d^\top \nabla f(x_k) \;\geq\; \mathrm{cm}(D_k)\,\|\nabla f(x_k)\| \geq \kappa_g\,\|\nabla f(x_k)\|.$$

Let d_k be a direction of D_k realizing this maximum. Then by considering this direction in (2.8), one obtains (2.4).

Consider now a direction $d \in V_k$. Note that relation (2.6) also holds when d is replaced by $-d$; if we sum the two resulting equations, we have:

$$-2\,\rho(\alpha_k) \;\leq\; f(x_k + \alpha_k\, d) + f(x_k - \alpha_k\, d) - 2\, f(x_k).$$

Thus, a second-order Taylor expansion of both $f(x_k + \alpha_k\, d)$ and $f(x_k - \alpha_k\, d)$ leads to:

$$-2\,\rho(\alpha_k) \;\leq\; \alpha_k^2\, d^\top \nabla^2 f(x_k)\, d + \frac{\nu_H}{3}\,\alpha_k^3, \tag{2.9}$$

thanks to the Lipschitz continuity of $\nabla^2 f$. In particular, (2.9) holds for the couple of directions that realize the minimum of $d^\top \nabla^2 f(x_k) d$ in V_k and the relation (2.5) can be easily derived. □

The previous result indicates that we can define a directional measure of second-order optimality that will play a similar role as the cosine measure in first-order optimality proofs. We thus introduce the *Rayleigh measure* of a set of vectors for a given matrix. This definition is inspired by the local cosine measure described in [25], which expresses the ability to approximate a particular vector, and by the Rayleigh quotient, a useful tool while dealing with second-order optimality [8]. Although not named, a similar quantity was used by Gratton et al. [26] in the case of trust-region methods.

Definition 2.1 *Let D be a set of unitary vectors in $\mathbb{R}^n$ and A an n-by-n real symmetric matrix. The* Rayleigh measure *of D with respect to A is given by:*

$$\mathrm{rm}\,(D, A) \;=\; \min_{d \in D}\; d^\top A\, d. \tag{2.10}$$

This measure is an approximation of the lowest eigenvalue of A by the minimum Rayleigh quotient among all vectors in D. This approximation is an exact one when D contains an eigenvector associated to the minimum eigenvalue of A: the Rayleigh measure is then equal to this lowest eigenvalue. More generally, if A has at least one negative eigenvalue, the Rayleigh measure indicates if the set D contains directions corresponding to negative curvature, i.e., to negative values of the Rayleigh quotient.

One sees that the Rayleigh measure of V_k with respect to the Hessian matrix appears in (2.5). This naturally encourages the use of this measure as an alternative to the minimum Hessian eigenvalue, which is the usual second-order criterion.

2.2 First-order and weak second-order global convergence results

In this section, we establish a second-order property related to the Rayleigh measures, hence to the partial curvature information we are able to collect at each iteration. This property generalizes the *pseudo-second order optimality* conditions presented for pattern search [1], and was called *weak second-order optimality* in [26] where the authors consider a trust-region framework with incomplete curvature information; we will use the latter terminology.

Whether they concern first-order or second-order optimality, most convergence proofs in direct-search methods are based on the convergence to zero of the step size along a subsequence of unsuccessful iterations. When sufficient decrease is imposed, it is known that the whole sequence of step sizes goes to zero. This is what is stated in the lemma below, which is proved for instance in [28].

Lemma 2.2 *Suppose that Assumption 2.2 holds. Then the step size sequence $\{\alpha_k\}$ produced by Algorithm 2.1 is such that*

$$\lim_{k\to\infty} \alpha_k \;=\; 0. \tag{2.11}$$

A first-order convergence proof uses (2.4) together with Lemma 2.2 to show that there exists a subsequence of iterates that drives the gradient to zero.

Since we are interested in establishing second-order properties, we go a step further and propose a weak second-order optimality criterion based on the sequence of Rayleigh measures. Note that such a technique does not guarantee that the algorithm is able to avoid converging to a maximizer or a saddle point. Abramson showed in [1] that Algorithm 2.1 may converge to critical points where the Hessian matrix has zero eigenvalues, even though those points are not local minimizers. However, for these examples, one still obtains the property called *weak second-order optimality* by Gratton et al. [26], that is a second-order optimality with respect to a set of directions. The formulation in our case is even more general, and is stated below.

Theorem 2.1 *Suppose that Assumptions 2.1–2.4 hold. Then*

$$\liminf_{k\to\infty}\, \max\left\{\|\nabla f(x_k)\|, -\operatorname{rm}\left(V_k, \nabla^2 f(x_k)\right)\right\} \;=\; 0. \tag{2.12}$$

In the specific case where $D_k = V_k$, the result becomes:

$$\liminf_{k\to\infty}\, \max\left\{\|\nabla f(x_k)\|, -\operatorname{rm}\left(D_k, \nabla^2 f(x_k)\right)\right\} \;=\; 0. \tag{2.13}$$

Proof. From Lemma 2.1, we know that for any unsuccessful iteration of index k,

$$\|\nabla f(x_k)\| \;\le\; \frac{1}{\kappa_g}\left[\frac{\rho(\alpha_k)}{\alpha_k} + \frac{\nu_g}{2}\alpha_k\right] \tag{2.14}$$

and

$$-\operatorname{rm}\left(V_k, \nabla^2 f(x_k)\right) \;\le\; \frac{2\rho(\alpha_k)}{\alpha_k^2} + \frac{\nu_H}{3}\,\alpha_k \tag{2.15}$$

hold, hence

$$\max\left\{\|\nabla f(x_k)\|, -\operatorname{rm}\left(V_k, \nabla^2 f(x_k)\right)\right\} \;\le\; \max\left\{\frac{1}{\kappa_g}\left[\frac{\rho(\alpha_k)}{\alpha_k} + \frac{\nu_g}{2}\alpha_k\right], \frac{2\rho(\alpha_k)}{\alpha_k^2} + \frac{\nu_H}{3}\,\alpha_k\right\}. \tag{2.16}$$

Lemma 2.2 ensures that there exists an infinite subsequence of unsuccessful iterations. For such a subsequence, both (2.14) and (2.15) hold, and the right part of each inequality goes to zero when k goes to infinity thanks to Assumption 2.3. We thus conclude that (2.12) holds. The specific case where $D_k = V_k$ is immediate. □

Our analysis shows that any direct-search method that follows the framework of Algorithm 2.1, i.e., uses a polling step and imposes a sufficient decrease of $o(\alpha^2)$ exhibits weak second-order properties. In practice, if we were to use the same symmetric set of directions at each iteration, we would know that, at any limit point, the corresponding Rayleigh measure is nonnegative. This result is tight in the sense that additional properties on the directions are needed to ensure that the method does not converge to a first-order stationary point that is not a minimum. Consider, for instance, applying Algorithm 2.1 to the following function presented in [3]:

$$f_1(x,y) \;=\; (9\,x-y)\,(11\,x-y)+\frac{x^4}{2}, \tag{2.17}$$

with $x_0 = (0,0)^\top$ as the initial point and $D_k = [e_1 \;\; e_2 \;\; -e_1 \;\; -e_2]$ for all k. One sees that the method cannot move away from the origin, which is a saddle point. In that case, the coordinate directions and their negatives are not of negative curvature, as the Rayleigh measure is equal to zero at each iteration; the method is thus weakly second-order convergent on this function, but not second-order globally convergent.

The following corollary clarifies the link between (2.12) and the second-order results based on limit of refining directions [1, 2, 3]. Note that such a result holds because of the continuity of the Rayleigh quotient.

Corollary 2.1 *Under the assumptions of Theorem 2.1, suppose that the sequence of iterates $\{x_k\}$ is bounded. Then there exists a subsequence of iterates $\{x_k\}_K$ converging to a limit point x_* such that $\nabla f(x_*) = 0$.*

Define the set of refining directions V_* by

$$V_* \;=\; \{d \in \mathbb{R}^n \mid \exists L \subset K, \{d_l\}_{l\in L} \to d, \forall l \in L, d_l \in D_l\}\,.$$

Then the curvature at x_* is nonnegative along the directions in V_*, i.e.,

$$\forall d \in V_*,\;\; d^\top\,\nabla^2 f(x_*)\,d \;\geq\; 0. \tag{2.18}$$

Note that if V_* is dense in the unit sphere, the limit point x_* is a second-order critical one; the second-order optimality is thus assured by a similar argument as for the MADS methods [2], although those algorithms do not enforce sufficient decrease.

PSSs without symmetric parts When one aims for first-order convergence results using PSSs, it is not necessary for those PSSs to have a non-empty symmetric part [28]. One might thus wonder if second-order results are still provable using PSSs for which the symmetric part is empty.

A feature of the directions of a given PSS $D = [d_1 \cdots d_m]$ is that there always exist m nonnegative scalars $(\beta_i)_{i=1,m}$ such that $\sum_{i=1}^m \beta_i\, d_i = 0$ [15]. Considering Algorithm 2.1 and using these scalars, one can proceed as in Lemma 2.1 to arrive at the following relation:

$$\sum_{i=1}^m \beta_i\, f(x_k+\alpha_k\, d_i) - \left(\sum_{i=1}^m \beta_i\right) f(x_k) \;\geq\; -\left(\sum_{i=1}^m \beta_i\right) \rho(\alpha_k), \tag{2.19}$$

which leads to

$$\sum_{i=1}^{m} \beta_i \, d_i^\top \nabla^2 f(x_k) d_i \;\geq\; -\sum_{i=1}^{m} \beta_i \left(\frac{\rho(\alpha_k)}{\alpha_k^2} + \frac{\nu_H}{6}\, \alpha_k \right). \tag{2.20}$$

We may then derive the analysis in a similar way as before, and obtain a result on the convergence of a weighted sum of Rayleigh quotients under the appropriate assumptions, that is:

$$\limsup_{k\to\infty} \sum_{i=1}^{|D_k|} \beta_i^{(k)} \, (d_i^{(k)})^\top \, \nabla^2 f(x_k) \, d_i^{(k)} \;\geq\; 0, \tag{2.21}$$

where, for all k, $\sum_i \beta_i^{(k)} \, d_i^{(k)} = 0$.

One notices that we obtain a weaker result than in the symmetric case; indeed, in (2.20), we used a specific positive combination $\{\beta_i\}$ without knowing if it is the most relevant one. Besides, the combination depends on the direction set, which possibly changes at each iteration, and the meaning of (2.21) is then unclear. When the sets are symmetric, however, we can explore this symmetry by constructing $|D|/2$ nonnegative combinations such that only the coefficients corresponding to a couple of opposite directions are not equal to zero (as we have seen in the proof of Lemma 2.1). The resulting properties are stronger as they involve the Rayleigh measure.

3 A direct-search method that ensures second-order optimality convergence

The goal of this section is to improve the second-order results of Section 2 in order to obtain a method that is second-order globally convergent in the usual sense. Ideally, we would like to define a second-order property on the polling directions which would be equivalent to the positive spanning property for the first order. In derivative-based methods, this is done by assuming that one of the directions is an approximate negative curvature one, i.e., that if the Hessian $\nabla^2 f(x_k)$ has a minimum eigenvalue $\lambda_k < 0$, that one of the directions $d \in D_k$ is such that

$$d^\top \nabla f(x_k) \;\leq\; 0 \quad \text{and} \quad d^\top \nabla^2 f(x_k)\, d \;\leq\; \kappa\, \lambda_k, \tag{3.1}$$

with $\kappa \in (0,1)$ independent of k. Such a requirement is classical in curvilinear line search methods [30, 32, 34] and second-order convergent line-search frameworks [23]. To generate such directions, one uses linear algebra techniques such as the Bunch-Parlett factorization coupled with a Krylov subspace method [23].

In a derivative-free context, we do not have access to the Hessian matrix or its product with a vector, but we can estimate Rayleigh quotients. The first part of (3.1) is easy to satisfy, but the second inequality poses a harder problem. Indeed, it can be rewritten as follows:

$$\mathrm{rm}(D_k, \nabla^2 f(x_k)) \;\leq\; \kappa \,\mathrm{rm}\left(\mathbb{S}^{n-1}, \nabla^2 f(x_k)\right) \;=\; \kappa \left(\min_{d \in \mathbb{S}^{n-1}} d^\top \nabla^2 f(x_k)\, d \right) \;=\; \kappa\, \lambda_k, \tag{3.2}$$

where $\mathbb{S}^{n-1}$ denotes the unit sphere in $\mathbb{R}^n$. It thus appears that a direct-search method attempts to estimate at each iteration the solution of a quadratic problem with a finite number of directions, and given the lack of knowledge on the quadratic form itself, it seems really demanding to ask for such an approximation.

At the same time, derivative-free trust-region methods have been proved to converge towards a second-order stationary point, provided the models used are sufficiently accurate (with error bounds as good as those provided by second-order Taylor models) and it is possible to compute an approximate eigenvector for the minimization of the model in the trust region [12]. We would like to combine features of the model-based methods with the existing direct-search techniques that use curvature estimation [3, 31].

3.1 Using a Hessian approximation to determine additional directions

The instance of Generating Set Search using Curvature Information presented in [20] uses a Hessian approximation which is updated along the iterations (the update may not occur at every iteration). In the unconstrained case, the new directions are then obtained by computing the eigenvectors of the approximate Hessian. It is then the quality of the approximation, of the order of the step size, that leads to second-order global convergence.

This approach requires the use of PSSs of the form $[Q \ -Q]$, where Q is an orthogonal matrix. However, it is known [15] that both positive spanning sets and positive bases are not necessarily made of $2n$ vectors, nor are they necessarily symmetric. We thus would like to extend the idea of [20] in a more general setting. In addition, we would like a method that does not completely overtakes the framework of Algorithm 2.1; the expense of searching for negative curvature should not intervene unless the usual first-order approach has failed. This is the second objective of our algorithm. Last but not least, the amount of function evaluations at each iteration should be of order of n^2, to be in accordance with the methods that build a Hessian approximation or use models with second-order accuracy.

The above requirements lead to the following direct-search instance, called AHDS, for Approximate Hessian Direct Search.

Algorithm 3.1 (Approximate Hessian Direct Search)

Init. *Choose* $x_0 \in \mathbb{R}^n$, $\alpha_{\max} > \alpha_0 > 0$, $0 < \theta < 1 < \gamma$, *and a* forcing function $\rho : \mathbb{R}^+ \to \mathbb{R}^+$. *Set* $k = 0$.

1. *Generate a Positive Spanning Set* D_k. *If there exists* $d \in D_k$ *such that*

$$f(x_k + \alpha_k \, d) \ < \ f(x_k) - \rho(\alpha_k), \tag{3.3}$$

   then declare iteration k *successful with the direction* $d_k = d$ *and go to 5.*
   Otherwise, go to 2.

2. *If there exists* $d \in D_k$ *such that* (3.3) *holds for* $-d$, *then declare the iteration successful with the direction* $d_k = -d$ *and go to 5. Otherwise, go to 3.*

3. *Choose* B_k *as a subset of* D_k *with* n *linearly independent directions, which we index as* $d_1, \cdots, d_n$.

   If there exists $d \in \{d_i + d_j, 1 \le i < j \le n\}$ *such that* (3.3) *holds, then declare the iteration successful with* $d_k = d$, *and go to 5. Otherwise go to 4.*

4. *Define the matrix* H_k *as*

$$(H_k)_{ii} \ = \ \frac{f(x_k + \alpha_k \, d_i) - 2\, f(x_k) + f(x_k - \alpha_k \, d_i)}{\alpha_k^2} \tag{3.4}$$

if $i = j$, and

$$(H_k)_{ij} = (H_k)_{ji} \;=\; \frac{f(x_k + \alpha_k\, d_i + \alpha_k\, d_j) - f(x_k + \alpha_k\, d_i) - f(x_k + \alpha_k\, d_j) + f(x_k)}{\alpha_k^2} \tag{3.5}$$

if $i \neq j$, with $(i,j) \in \{1,\ldots,n\}^2$. Compute a unitary eigenvector v_k associated with the minimum eigenvalue of H_k. If v_k or $-v_k$ satisfies the decrease condition (3.3), declare the iteration successful with d_k equal to v_k or $-v_k$ (depending on which vector yields the lowest function value), otherwise declare the iteration unsuccessful and go to 5.

5. *If the iteration was declared successful, set $x_{k+1} = x_k + \alpha_k\, d_k$, $\alpha_{k+1} = \min\{\gamma\, \alpha_k, \alpha_{\max}\}$. Otherwise set $x_{k+1} = x_k, \alpha_{k+1} = \theta\, \alpha_k$.*

6. *Increment k by one and go to 1.*

Algorithm 3.1 is close in spirit to the superlinearly convergent method developed by Mifflin [31], although we do not use a gradient approximation. Here, we rather focus on exploiting negative curvature if possible.

Note that in case of a successful iteration at Step 1, the method behaves like Algorithm 2.1 with a PSS. Note also that we always require a decrease using $\rho(\alpha_k)$ whether the directions are unitary or not, but this does not affect the convergence nor the complexity analyses.

3.2 Second-order global convergence of the new method

Having presented our method, we now show that it is indeed second-order globally convergent. The proof requires two intermediate results, that respectively enlighten the properties of the approximate eigenvector v_k and the theoretical guarantees of every unsuccessful iteration.

Proposition 3.1 *Let v_k be the unitary vector described in Step 4 of Algorithm 3.1. Suppose that f satisfies Assumption 2.1 and that $\lambda_{\min}(\nabla^2 f(x_k)) < 0$. Then one has:*

$$v_k^\top\, \nabla^2 f(x_k)\, v_k \;\le\; \sigma_{\min}(B_k)^2\, \lambda_k + \frac{10\, \nu_H\, n}{3}\alpha_k. \tag{3.6}$$

where $\lambda_k = \lambda_{\min}(\nabla^2 f(x_k))$.

Proof. The formulas defining the approximated Hessian H_k together with f satisfying Assumption 2.1 lead to the following error bound:

$$\forall (i,j) \in \{1,\ldots,n\}^2,\; \left|(H_k)_{ij} - d_i^\top\, \nabla^2 f(x_k)\, d_j\right| \;\le\; \frac{5\, \nu_H}{3}\alpha_k, \tag{3.7}$$

hence

$$\left\|H_k - B_k^\top\, \nabla^2 f(x_k)\, B_k\right\| \;\le\; \left\|H_k - B_k^\top\, \nabla^2 f(x_k)\, B_k\right\|_F \;\le\; n\,\frac{5\, \nu_H}{3}\alpha_k, \tag{3.8}$$

where $\|\cdot\|$ denotes the Euclidean norm and $\|\cdot\|_F$ the Frobenius norm. From this bound on the approximation error, one obtains a bound regarding the minimum eigenvalue approximation (see [13, Proposition 10.14]):

$$\left|\lambda_{\min}(H_k) - \lambda_{\min}\left(B_k^\top\, \nabla^2 f(x_k)\, B_k\right)\right| \;\le\; n\,\frac{5\, \nu_H}{3}\alpha_k. \tag{3.9}$$

Putting all together, one obtains

$$\begin{aligned}
v_k^\top \nabla^2 f(x_k)\, v_k &= \lambda_{\min}(H_k) + v_k^\top \left[\nabla^2 f(x_k) - H_k\right] v_k \\
&\le \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) + \left|\lambda_{\min}(H_k) - \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right)\right| + \\
&\quad \|v_k\|^2 \, \frac{5\, n\, \nu_H}{3}\, \alpha_k \\
&\le \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) + (1 + \|v_k\|^2)\, \frac{5\, n\, \nu_H}{3}\, \alpha_k \\
&= \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) + \frac{10\, n\, \nu_H}{3}\, \alpha_k.
\end{aligned}$$

Since B_k is a basis of $\mathbb{R}^n$, $B_k^\top B_k$ is positive definite. For every vector $y \in \mathbb{R}^n \setminus \{0\}$, we have

$$\begin{aligned}
\frac{y^\top B_k^\top \nabla^2 f(x_k)\, B_k\, y}{\|y\|^2} &\ge \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right), \\
\frac{y^\top B_k^\top \nabla^2 f(x_k)\, B_k\, y}{\|y\|^2} \times \frac{\|y\|^2}{y^\top B_k^\top B_k\, y} &\ge \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) \frac{\|y\|^2}{y^\top B_k^\top B_k\, y}, \\
\frac{y^\top B_k^\top \nabla^2 f(x_k)\, B_k\, y}{y^\top B_k^\top B_k\, y} &\ge \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) \frac{\|y\|^2}{y^\top B_k^\top B_k\, y}.
\end{aligned}$$

Taking the minimum over all non-zero vectors in $\mathbb{R}^n$, one obtains:

$$\lambda_{\min}(\nabla^2 f(x_k)) \;\ge\; \lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) \max_{y \neq 0} \frac{\|y\|^2}{y^\top B_k^\top B_k\, y}, \tag{3.10}$$

again using the fact that B_k is a basis. Indeed, this ensures that both minimum eigenvalues have the same sign: consequently, $\lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right) < 0$ and the minimum becomes a maximum. One finally has

$$\lambda_{\min}(\nabla^2 f(x_k)) \;\ge\; \frac{\lambda_{\min}\left(B_k^\top \nabla^2 f(x_k)\, B_k\right)}{\sigma_{\min}(B_k)^2}, \tag{3.11}$$

hence the result. □

Note that in Algorithm 3.1, we do not allow for a computation of the approximate Hessian along several iterations, yet in such cases, one can still derive errors bounds that turn out to be worse than those presented above. Indeed, in a scenario where an approximation of the Hessian is computed separately along p successive iterations, one can prove that if such iterations are unsuccessful, then the error bound (3.8) becomes of order $\mathcal{O}(\theta^{-p}\, n\, \nu_H\, \alpha_k)$. This holds for a naive implementation of the method, thus these bounds are likely to be improved by considering efficient practical strategies. However, this would require a thorough study that is beyond the scope of the present paper.

Proposition 3.1 shows that the approximation error between $v_k^\top \nabla^2 f(x_k)\, v_k$ and λ_k involves the minimum singular value of a certain matrix, as well as an error of order $\mathcal{O}(n\, \nu_H\, \alpha_k)$. These elements are consistent with those obtained when using fully quadratic models (see [13, Part I] and the references therein). In fact, the square of the singular value $\sigma_{\min}(B_k)$ plays a role that is similar to the poisedness constant, hence we make the following assumptions on those singular values.

Assumption 3.1 *The polling sets satisfy Assumption 2.4. In addition, the bases B_k are chosen such that there exists $\sigma > 0$, independent of k, such that*

$$\forall k, \quad \sigma_{\min}(B_k)^2 \geq \sigma. \tag{3.12}$$

When the B_k are orthonormal bases, one can choose $\sigma = \sigma_{\min}(B_k) = 1$. This is the case, for instance, when all polling sets are equal to $[Q \ -Q]$, with Q being an orthogonal matrix.

Lemma 3.1 *Consider an unsuccessful iteration of Algorithm 3.1 such that $\mathrm{cm}(D_k) \geq \kappa_g > 0$ and $\sigma_{\min}(B_k)^2 \geq \sigma$. Suppose that f satisfies Assumption 2.1. In that case,*

$$\|\nabla f(x_k)\| \ \leq \ \kappa_g^{-1} \left(\frac{\rho(\alpha_k)}{\alpha_k} + \frac{\nu_g}{2} \alpha_k \right) \tag{3.13}$$

is satisfied, and, if $\lambda_k < 0$,

$$\lambda_k \ \geq \ -\sigma^{-1} \left(\frac{2\,\rho(\alpha_k)}{\alpha_k^2} + (10\,n + 1) \frac{\nu_H}{3} \alpha_k \right) \tag{3.14}$$

holds.

Proof. Equation (3.13) is obtained as in the proof of Lemma 2.1, considering that we use a PSS D_k at each iteration.

To arrive at (3.14), notice that we evaluate f at both $x_k + \alpha_k\, v_k$ and $x_k - \alpha_k\, v_k$. Thus, we can obtain the analogous of (2.9) for v_k, which is

$$-2\,\rho(\alpha_k) \ \leq \ \alpha_k^2\, v_k^\top\, \nabla^2 f(x_k)\, v_k + \frac{\nu_H}{3} \alpha_k. \tag{3.15}$$

Since we are in the assumptions of Proposition 3.1, we can replace the Rayleigh quotient by an expression only depending on λ_k and α_k, and we arrive at (3.14). □

We point out that for unsuccessful iterations, the corresponding Rayleigh measure is an approximation of the minimum eigenvalue with an error in $\mathcal{O}(\alpha_k)$: this is the key property that turns the weak second-order results into strong second-order ones. Indeed, we obtain the following convergence theorem, whose proof follows the one of Theorem 2.1, only with λ_k playing the role of $\mathrm{rm}\left(V_k, \nabla^2 f(x_k)\right)$.

Theorem 3.1 *We consider Algorithm 3.1 under Assumptions 2.1, 2.2, 2.3 and 3.1. Then,*

$$\liminf_{k \to \infty} \max \left\{ \|\nabla f(x_k)\|, -\lambda_k \right\} \ = \ 0, \tag{3.16}$$

i.e., the method is second-order globally convergent.

This result confirms that whenever directions determined by a Hessian approximation are used, the accuracy of the approximation is the key for controlling the second-order criterion (namely the minimum Hessian eigenvalue). This has an undeniable cost in terms of function evaluations. However, if some special structure is known about the problem, then this amount of evaluations can be reduced. Random sampling can also reduce the cost if the Hessian is sparse, even if the sparsity pattern is unknown [6].

4 Worst case complexity

This section is dedicated to the complexity of direct search in determining second-order stationary points. We mainly develop our reasoning for the "strong" second-order globally convergent approach of Section 3.2. This being said, the upcoming analysis is valid for any second-order criterion of interest, and weak second-order complexity results are also discussed at the end.

We are looking for a bound on the number of iterations needed to ensure:

$$\inf_{0 \le l \le k} \|\nabla f(x_l)\| < \epsilon_g \text{ and } \sup_{0 \le l \le k} \lambda_k > -\epsilon_H, \tag{4.1}$$

given two thresholds $\epsilon_g, \epsilon_H \in (0,1)$. When (4.1) is satisfied, we say that we reached approximate second-order optimality, with respect to ϵ_g and ϵ_H. Although the first-order result established by Vicente [36, Corollary 3.1] could still be applied to the settings of Sections 2 and 3, we treat first and second-order optimality simultaneously, for both self-containedness and clarity. The analysis establishes separate bounds on the number of successful and unsuccessful iterations needed to achieve (4.1).

For the rest of this section, we will consider a typical family of forcing functions, namely $\rho(t) = \frac{c}{6}\, t^p$, where $c > 0$ and $p > 2$ (those functions clearly satisfy Assumption 2.3). We start by bounding the number of successful iterations.

Theorem 4.1 *Suppose that the assumptions of Theorem 3.1 hold. Assume that Algorithm 3.1 is applied with $\rho(t) = \frac{c}{6}\, t^p$ with $c > 0$ and $p > 2$.*

Given $\epsilon_g, \epsilon_H \in (0,1)$, let k_0 be the index of the first unsuccessful iteration and assume that (4.1) does not hold, and let l_1 be the first index such that (4.1) is satisfied at iteration $l_1 + 1$. Then the number of successful iterations between k_0 and l_1, denoted by $|S_{l_1}(k_0)|$, is bounded as follows:

$$|S_{l_1}(k_0)| \;\le\; \left\lceil \left(\frac{6(f(x_{l_0}) - f_{low})}{c\, \theta^p\, L_s^p} \right) \max\left(\kappa_g^{-p}\, \epsilon_g^{-p}, (\sigma^{-1}\, n)^{\frac{p}{\min(p-2,1)}}\, \epsilon_H^{-\frac{p}{\min(p-2,1)}} \right) \right\rceil, \tag{4.2}$$

where

$$L_s = \min\left(1, L_1^{-1}, L_2^{-\frac{1}{\min(p-2,1)}}\right), \; L_1 = \frac{c + 3\,\nu_g}{6}, \text{ and } L_2 = \frac{c + 11\,\nu_H}{3}.$$

Proof. For every $l \in U$ such that $k_0 \le l < l_1$, we know that (4.1) does not hold, and thus either

$$\|\nabla f(x_l)\| \;\ge\; \epsilon_g \tag{4.3}$$

or

$$\lambda_l \;\le\; -\epsilon_H. \tag{4.4}$$

In the first case, using (3.13), we have that

$$\epsilon_g \;\le\; \|\nabla f(x_l)\| \;\le\; \kappa_g^{-1} \left[\frac{c}{6}\, \alpha_l^{p-1} + \frac{\nu_g}{2} \alpha_l \right].$$

Thus, if $\alpha_l < 1$,

$$\epsilon_g \;\le\; \frac{c + 3\,\nu_g}{6\,\kappa_g}\, \alpha_l^{\min(p-1,1)}$$

and if not $\alpha_l \geq 1 > \epsilon_g$, from which we deduce

$$\alpha_l \;\geq\; \min(1, L_1^{-\frac{1}{\min(p-1,1)}})\, \kappa_g^{\frac{1}{\min(p-1,1)}}\, \epsilon_g^{\frac{1}{\min(p-1,1)}}.$$

Since $p > 2$, this reduces to

$$\alpha_l \;\geq\; \min(1, L_1^{-1})\, \kappa_g\, \epsilon_g. \tag{4.5}$$

In the second case, we obtain from (3.14) that

$$-\epsilon_H \;\geq\; \lambda_l \;\geq\; -\sigma^{-1}\left(\frac{c}{3}\,\alpha_l^{p-2} + \frac{(10\,n+1)\,\nu_H}{3}\,\alpha_l\right) \;\geq\; -\sigma^{-1}\,n\,\left(\frac{c}{3}\,\alpha_l^{p-2} + \frac{11\,\nu_H}{3}\,\alpha_l\right),$$

which leads by the same reasoning as above to

$$\alpha_l \;\geq\; \min(1, L_2^{-\frac{1}{\min(p-2,1)}})\,(\sigma^{-1}\,n)^{-\frac{1}{\min(p-2,1)}}\,\epsilon_H^{\frac{1}{\min(p-2,1)}}, \tag{4.6}$$

As a result of (4.5) and (4.6), for all unsuccessful iterations of index $k_0 \leq l < l_1$, one has the following lower bound on the step size

$$\alpha_l \;\geq\; L_s\,\min\left(\kappa_g\,\epsilon_g, (\sigma^{-1}\,n)^{-\frac{1}{\min(p-2,1)}}\,\epsilon_H^{\frac{1}{\min(p-2,1)}}\right).$$

Consider now the successful iterations of index k, $k_0 < k \leq l_1$. For each iteration k of this type, one can backtrack to the previous unsuccessful iteration (which exists since $k_0 \in U$), denoted by $l(k)$, such that $\alpha_k \geq \theta\,\alpha_{l(k)}$, given the update rules for the step size parameter. Thus, for any of those iterations, one has:

$$\alpha_k \;\geq\; \theta\,L_s\,\min\left(\kappa_g\,\epsilon_g, (\sigma^{-1}\,n)^{-\frac{1}{\min(p-2,1)}}\,\epsilon_H^{\frac{1}{\min(p-2,1)}}\right), \tag{4.7}$$

and by definition of a successful iteration:

$$\begin{aligned} f(x_k) - f(x_{k+1}) \;&\geq\; \rho(\alpha_k) \\ &\geq\; \frac{c}{6}\,(\theta\,L_s)^p\,\min\left(\kappa_g^p\,\epsilon_g^p, (\sigma^{-1}\,n)^{-\frac{p}{\min(p-2,1)}}\,\epsilon_H^{\frac{p}{\min(p-2,1)}}\right). \end{aligned}$$

Thus, by summing on all successful iterations until l_1 excluded, we arrive at

$$f(x_{k_0}) - f(x_{l_1}) \;\geq |S_{l_1}(k_0)|\,\frac{c}{6}\,(\theta\,L_s)^p\,\min\left(\kappa_g^p\,\epsilon_g^p, (\sigma^{-1}\,n)^{-\frac{p}{\min(p-2,1)}}\,\epsilon_H^{\frac{p}{\min(p-2,1)}}\right).$$

and the result stated in the theorem follows from Assumption 2.2. □

We then treat the case of the unsuccessful iterations.

Theorem 4.2 *Let the assumptions of Theorem 4.1 hold. Then, with the same definitions for k_0, l_1, the number of unsuccessful iterations between k_0 and l_1 is at most $|U_{l_1}(k_0)|$, where*

$$|U_{l_1}(k_0)| \;\leq\; \left\lceil L_3\,|S_{l_1}(k_0)| + L_4 - \log_\theta e\,\max\left\{\kappa_g^{-1}\,\epsilon_g^{-1}, (\sigma^{-1}\,n)^{\frac{1}{\min(p-2,1)}}\,\epsilon_H^{-\frac{1}{\min(p-2,1)}}\right\}\right\rceil \tag{4.8}$$

with

$$L_3 = -\log_\theta \gamma \quad \textit{and} \quad L_4 = \log_\theta\left(\frac{\theta\,L_s\,e}{\alpha_{k_0}}\right).$$

Proof. By induction, one has:

$$\alpha_{l_1} \;\leq\; \alpha_{k_0}\, \gamma^{|S_{l_1}(k_0)|}\, \theta^{|U_{l_1}(k_0)|},$$

which, as $\theta \in (0,1)$, leads to

$$|U_{l_1}(k_0)| \;\leq\; -\log_\theta \gamma\, |S_{l_1}(k_0)| - \log_\theta \alpha_{k_0} + \log_\theta \alpha_{l_1}. \tag{4.9}$$

Since $\ln\theta < 0, \ln\gamma > 0$, and α_{l_1} is bounded below from (4.7), (4.9) becomes

$$|U_{l_1}(k_0)| \;\leq\; L_3\, |S_{l_1}(k_0)| + \log_\theta\left(\frac{\theta\, L_s}{\alpha_{k_0}}\right) - \frac{\ln\left(\max\left\{\kappa_g^{-1}\,\epsilon_g^{-1}, (\sigma^{-1}\,n)^{\frac{1}{\min(p-2,1)}}\,\epsilon_H^{-\frac{1}{\min(p-2,1)}}\right\}\right)}{\ln\theta}.$$

Finally, we apply $\ln(x) \leq x - 1$ and arrive at the desired result. □

As explained in [36], the index of the first unsuccessful iteration k_0 can be bounded from above. In our case, we can choose the following quantity as an upper bound:

$$\left\lceil \frac{6(f(x_0) - f_{low})}{c\,\alpha_0^p}\, \max\left(\kappa_g^{-p}\,\epsilon_g^{-p}, (\sigma^{-1}\,n)^{\frac{p}{\min(p-2,1)}}\,\epsilon_H^{-\frac{p}{\min(p-2,1)}}\right)\right\rceil.$$

This leads to the following result regarding weak second-order optimality.

Theorem 4.3 *Let the assumptions of Theorem 4.1 hold. The number of iterations needed by Algorithm 3.1 to satisfy* (4.1) *is at most*

$$\mathcal{O}\left(\max\left(\kappa_g^{-p}\,\epsilon_g^{-p}, (\sigma^{-1}\,n)^{\frac{p}{\min(p-2,1)}}\,\epsilon_H^{-\frac{p}{\min(p-2,1)}}\right)\right), \tag{4.10}$$

where the constant in $\mathcal{O}(\cdot)$ *depends on* $\nu_g, \nu_H, \alpha_0, f(x_0), f_{low}, c, \gamma, \theta,$ *and* p.

The best power of ϵ_H (that is, the least negative power) is here achievable choosing $p = 3$.

We now give the corresponding result with respect to the number of function evaluations.

Theorem 4.4 *Under the assumptions of Theorem 4.1, the number of function evaluations needed by Algorithm 3.1 is at most*

$$\mathcal{O}\left(m\,\max\left(\kappa_g^{-p}\,\epsilon_g^{-p}, (\sigma^{-1}\,n)^{\frac{p}{\min(p-2,1)}}\,\epsilon_H^{-\frac{p}{\min(p-2,1)}}\right)\right), \tag{4.11}$$

with m *is the maximum number of function evaluations performed in any iteration. Here again, the constant in* $\mathcal{O}(\cdot)$ *only depends on* $\nu_g, \nu_H, \alpha_0, f(x_0), f_{low}, c, \gamma, \theta,$ *and* p.

As in the first-order case [11, 17, 21, 36], we are interested in the order of n that appears in the complexity bounds related to the number of function evaluations. We will see that such an order is considerably higher than in the first-order case, which is not surprising given the requirements we impose on the polling sets.

The value of m in (4.11) depends on the choice of the polling sets, their cardinality and whether they have a non empty symmetric part. For instance, $D_k = D_\oplus = [I\ -I]$ leads to at most

$$m = 2\,n + \frac{n^2 - n}{2} + 2 = \frac{n^2 + 3\,n + 4}{2}$$

evaluations. In addition, one can only obtain orthonormal bases from $D_\oplus$ and $\mathrm{cm}(D_\oplus) = 1/\sqrt{n}$ and thus one can replace κ_g by $1/\sqrt{n}$ and σ by 1. However, when $p = 3$, κ_g^{-p} becomes $n^{\frac{3}{2}}$, thus less than n^3, showing that the second-order part dominates the power of n in (4.10). The dependence of (4.11) on n, when using $D_\oplus$, is of the order n^5.

Corollary 4.1 *Consider the application of Algorithm 3.1, under the assumptions of Theorem 4.1 with $p = 3$. Suppose D_k is chosen as $D_\oplus$ for all k (or as any other PSS D such that $m = \mathcal{O}(n^2)$, $\mathrm{cm}(D) = \mathcal{O}\left(1/\sqrt{n}\right)$, and the B_k's are orthogonal matrices). Then, to satisfy* (4.1), *the method takes at most*

$$\mathcal{O}\left(n^3 \max\left\{\epsilon_g^{-3}, \epsilon_H^{-3}\right\}\right) \tag{4.12}$$

iterations and

$$\mathcal{O}\left(n^5 \max\left\{\epsilon_g^{-3}, \epsilon_H^{-3}\right\}\right) \tag{4.13}$$

function evaluations, where the constant in $\mathcal{O}(\cdot)$ only depends on $\nu_g, \nu_H, \alpha_0, f(x_0), f_{low}, c, \gamma$, and θ.

As explained in Section 3, preliminary knowledge regarding the structure of the Hessian may help reducing the powers of n.

Our analysis covers a wide class of direct-search algorithms. Note that it can be simplified following the process of Konečný and Richtárik [29] in the case where the step size is never increased and it is halved at every unsuccessful iteration (i.e., $\gamma = 1$ and $\theta = 1/2$). Choosing $\rho(t) = t^3$ (hence $p = 3$) as well as $D_k = D_\oplus$ for all iterations (as they provide the best known bounds), one could easily see that the number of successful iterations becomes:

$$|S_{l_1}(k_0)| \;\le\; \left\lceil \left(\frac{f(x_{l_0}) - f_{low}}{8\, L_s^3}\right) \max\left(n^{\frac{3}{2}}\, \epsilon_g^{-3}, n^3\, \epsilon_H^{-3}\right) \right\rceil, \tag{4.14}$$

by the same reasoning as in the proof of Theorem 4.1. On the other hand, since $\gamma = 1$ it would be much simpler to bound the number of unsuccessful iterations. The result corresponding to (4.8) is

$$|U_{l_1}(k_0)| \;\le\; \left\lceil \log_2(\alpha_{k_0}) - \log_2\left(\frac{L_s}{2} \min\left\{\frac{\epsilon_g}{\sqrt{n}}, \frac{\epsilon_H}{n}\right\}\right) \right\rceil. \tag{4.15}$$

The conclusions of Corollary 4.1 are then unchanged in terms of dependences on n,ϵ_g, and ϵ_H.

To the best of our knowledge, the above results are the first complexity bounds to be established regarding the determination of second-order stationary points by a derivative-free method. It is interesting to compare them to the existing second-order complexity bounds that have been obtained in the derivative-based literature. Cartis et al [10] derived such bounds for ARC and trust-region methods, respectively in

$$\mathcal{O}\left(\max\left\{\epsilon_g^{-\frac{3}{2}}, \epsilon_H^{-3}\right\}\right) \quad \text{and} \quad \mathcal{O}\left(\max\left\{\epsilon_g^{-2}\epsilon_H^{-1}, \epsilon_H^{-3}\right\}\right).$$

They proved also that whenever $\epsilon_H \le \epsilon_g$, the two bounds reduce to $\mathcal{O}(\epsilon_H^{-3})$, and gave an example to show that such a bound is sharp in terms of this tolerance. Our bound also reduces to $\mathcal{O}(\epsilon_H^{-3})$ in that case, so we may say that the three bounds are comparable from such a point of view; in our case, the sharpness of the bound remains to be proved.

When keeping both tolerances ϵ_g and ϵ_H in the bounds, we see that the first part of our bound is worse than the one obtained by ARC, whereas it seems comparable to the one for trust-region methods. However, we obtain ϵ_g^{-3} instead of $\epsilon_g^{-2}\,\epsilon_H^{-1}$. This discrepancy is related to the decrease requirements. In trust-region methods, provided (4.1) is not satisfied, one has

$$f(x_k) - f(x_{k+1}) \;\geq\; \eta \,\min\left\{\mu_1\,\epsilon_g\,\delta_k, \mu_2\,\epsilon_H\,\delta_k^2\right\} \quad \text{and} \quad \delta_k \geq \mu_3\,\min\{\epsilon_g, \epsilon_H\}, \tag{4.16}$$

where $\eta, \mu_1, \mu_2, \mu_3$ are positive constants and δ_k is the trust-region radius, which is the key argument to prove the complexity results [10, Section 4]. Two decreases are considered in (4.16), concerning respectively first and second-order criteria. In our direct-search frameworks, we only have one decrease formula that depends on α_k^3, hence the discrepancy.

Although we were not able to find second-order complexity results for derivative-free equivalents of ARC and trust-region methods [11, 12] in the existing literature, it is our belief that these results would match those in [10] regarding the powers of the tolerances, again because of the decrease formulas that are employed to prove the convergence.

Finally, we point out that a complexity result can be established given a weak second-order criterion such as the one presented in Section 2. The resulting bounds are of the same order in terms of powers of ϵ_g and ϵ_H, but, due to the fact that we use the result of Lemma 2.1 to establish the complexity bounds, one can reduce the dependence of the dimension up to

$$\mathcal{O}(n^{\frac{3}{2}}\,\max\left\{\epsilon_g^{-3}, \epsilon_H^{-3}\right\}) \tag{4.17}$$

on the number of iterations and

$$\mathcal{O}(n^{\frac{5}{2}}\,\max\left\{\epsilon_g^{-3}, \epsilon_H^{-3}\right\}) \tag{4.18}$$

for the number of function evaluations. Although these results concern weaker properties, they can serve for a comparison between our general direct-search method and, say, a trust-region method with incomplete curvature information such as the one developed in [26]. For such a trust-region scheme, it may be possible to derive a complexity result, either in a derivative-based [27] or a derivative-free [21] context.

5 Numerical observations

Having introduced second-order aspects in a direct-search framework, we have established that such a method converges towards second-order stationary points. This ensures that the algorithm is able to exploit negative curvature whenever needed. For instance, we can apply Algorithm 3.1 to the function f_1 described in Section 2.2, starting at the origin with the polling set equal to $D_\oplus$. As we have seen, none of these polling directions will yield a decrease in the objective function, because the corresponding values in those directions are always positive. However, if we compute an approximate Hessian and its eigenvector associated with the minimum eigenvalue, we obtain a direction in which the function has a negative value. With a sufficiently small step size, it will satisfy the sufficient decrease condition.

Besides escaping saddle points and local maximizers, the use of negative curvature is also known to improve the practical performance of line-search methods [23, 30, 34], as going in negative curvature directions possibly leads to faster decrease in the objective function value. We are interested in knowing if our approach allows for the same kind of improvement.

We compare implementations of Algorithms 2.1 and 3.1 on less pathological cases than the one mentioned above. The test functions are taken from the CUTEst package [24], and have been identified as presenting negative curvature at some points by Avelino et al. [5, Table 6].

This represents a total of 60 problems out of the 119 problems tested in [5]. For all of those problems, we used the smallest dimension available in the SIF files, resulting in 36 problems with dimensions less than 10, 22 problems having dimensions between 10 and 15, 1 problem with dimension 28, and 1 problem with dimension 50.

We tested four types of polling sets choices for Step 1 in Algorithm 3.1. The two first choices correspond to the set $D_\oplus$, that we defined in Section 4, and a minimal positive basis with uniform angles (see [13, Corollary 2.6]), denoted by V_{n+1}. These are common choices in practice, and also typical examples of polling sets, respectively with and without symmetric parts. The two other types of PSSs are based on $D_\oplus$ and V_{n+1}, built by applying a rotation matrix Q to those sets. All of our choices keep D_k constant throughout all iterations. However, to measure the effect of our approach on a larger variety of sets, we ran the method ten times, varying the ordering of the vectors for the choices $\{D_\oplus, V_{n+1}\}$, or changing the rotation matrix Q in the other cases. Table 1 summarizes our polling set choices for clarity; in the rest of the section, we will identify a method as bds$i$ or ahds$i$ with $i \in \{0, 1, 2, 3\}$ indicating the polling choice.

Polling Number	Set type	Cardinality	Variant
0	$D_\oplus$	$2\,n$	Ordering of directions
1	$Q\,D_\oplus$	$2\,n$	Rotation matrix Q
2	V_{n+1}	$n+1$	Ordering of directions
3	$Q\,V_{n+1}$	$n+1$	Rotation matrix Q

Table 1: The different polling set choices.

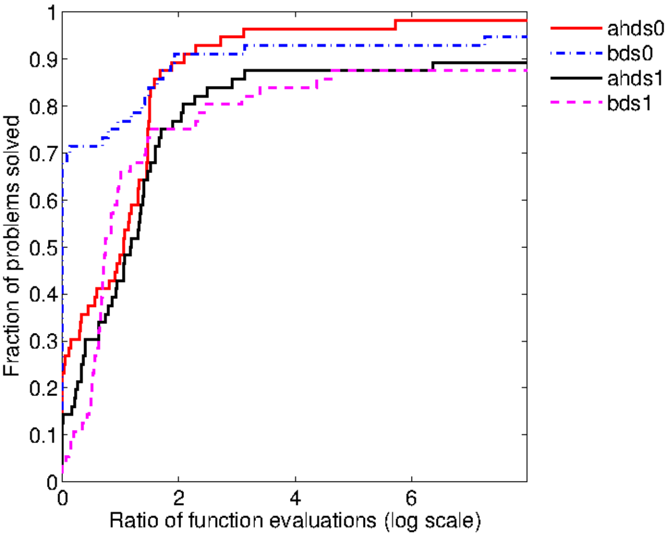
For all methods, the forcing function was $\rho(t) = 10^{-3}\,t^3$, the starting point was the one defined by CUTEst, and the starting step size was $\alpha_0 = 1$. We consider that a run is successful whenever the final function value f_* satisfies:

$$f_* - f_{opt} < \epsilon\,\left(f(x_0) - f_{opt}\right), \tag{5.1}$$

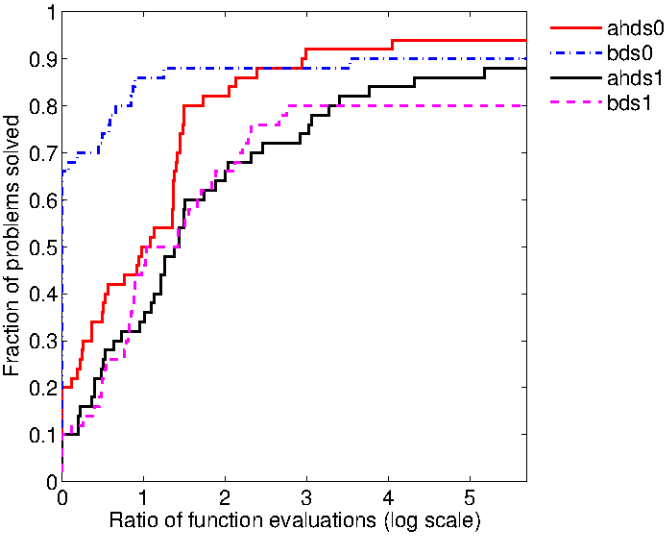
where f_{opt} is the best value obtained by all variants with an extended budget of $5000\,n$ iterations, and $\epsilon > 0$ is a given tolerance [33]. For each method, it is plotted a performance profile [18] for the average number of function evaluations taken on the 10 runs.

The results we present illustrate the typical behavior of Algorithm 3.1 compared to the classic direct-search scheme of Algorithm 2.1. Figure 1 firstly shows profiles obtained for the symmetric polling choices 0/1. One sees that the methods bds0 and bds1 perform still better than ahds0 and ahds1 in terms of efficiency (ratio=0). However, the ahds methods eventually solve more problems than the bds ones (large ratio), thus being more robust. This tendency was to be expected, as second-order mechanisms help in exploiting negative curvature.

Figure 2 is related to the non-symmetric polling set choices, and is characteristic of the performances of both methods. For such polling strategies, the gain from the bds to the ahds methods is even higher than in the symmetric case and now also present in efficiency. However, when one looks at the directions that lead to successful iterations for the ahds methods, one sees that half of the successful iterations are successful at Step 1 of Algorithm 3.1, one third are successful at Step 2, and only around 10% at Step 4. Our interpretation is that considering opposite directions already increases the chances to exploit negative curvature (as it guarantees weak second-order global convergence), while allowing to poll along additional directions in the case of a non-symmetric PSS.



(a) $\epsilon = 10^{-3}$.

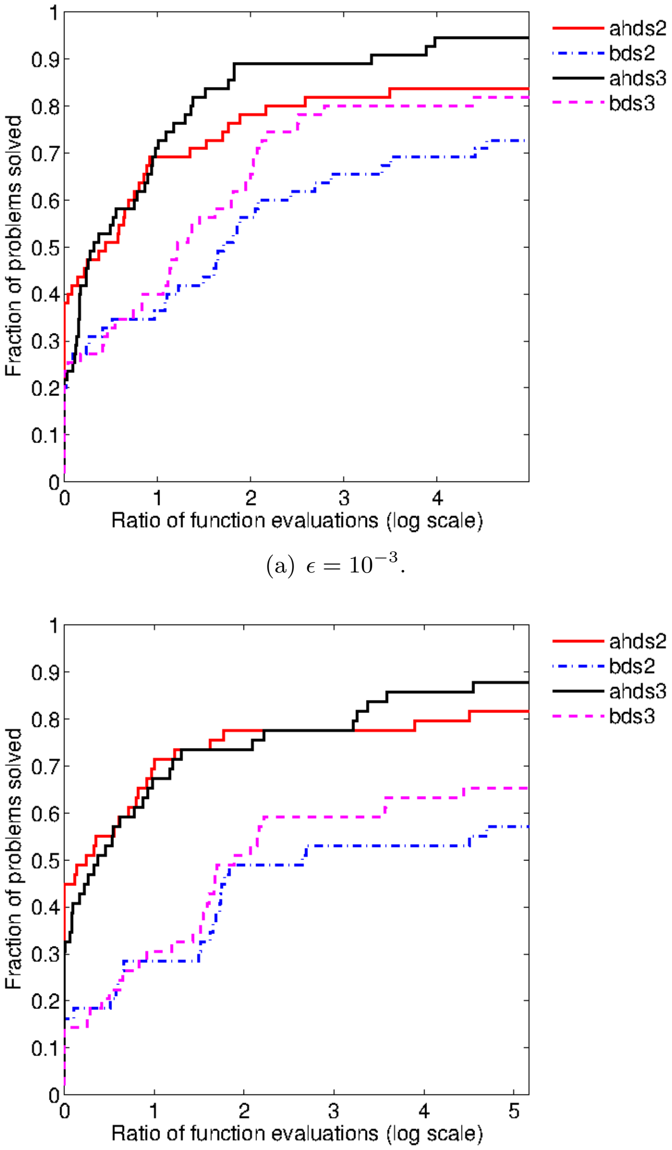


(b) $\epsilon = 10^{-6}$.

Figure 1: Performance of the methods with polling choices 0/1, given a budget of $2000\,n$ evaluations.

To support this hypothesis, we implemented a variant of Algorithm 3.1 without Steps 3 and 4; this method will be denoted by `sds` (symmetrized direct search) in the rest of the section. We aim to compare `bds`, `ahds`, and `sds` algorithms using non-symmetric PSSs for polling. For such a polling choice, the `bds` method only exhibits first-order properties, while the `sds` variant is weakly second-order convergent in the sense of Theorem 2.1. As for the `ahds` method, it is second-order convergent.

Figures 3 and 4 present the results for the polling strategies 2/3. One observes that the `sds` method is generally more efficient than the corresponding `bds` and `ahds` instances, the

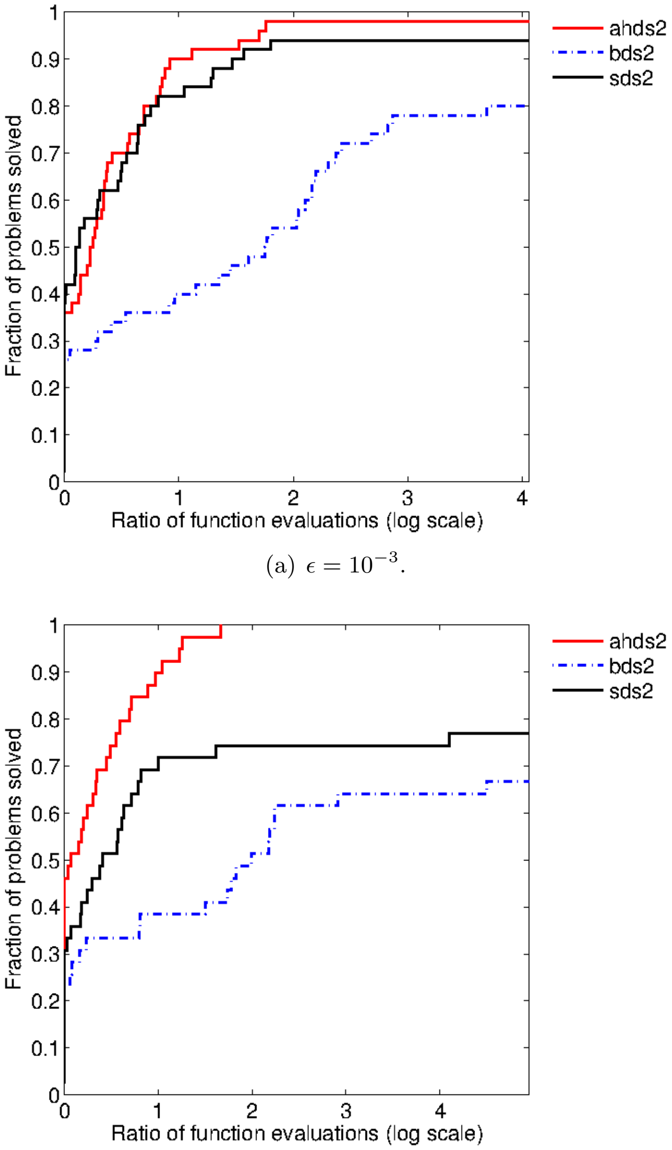


(a) $\epsilon = 10^{-3}$.

(b) $\epsilon = 10^{-6}$.

Figure 2: Performance of the methods with polling choices 2/3, given a budget of $2000\,n$ evaluations.

only exception being when $\epsilon = 10^{-6}$ and $D_k = V_{n+1}$. In this particular setting, the ahds method outperforms the other two, and the reason appears to be that the amount of successful iterations corresponding to the Step 4 of the method is significantly higher than in the other settings. Besides, on both Figures 3 and 4, the ahds method clearly stands out as the most robust implementation. The conclusions of Figure 1 can thus be extended for the non-symmetric case: the ahds algorithm eventually benefits from the computation of an approximate Hessian eigenvector. These profiles promote the use of symmetric positive spanning sets as a first attempt to catch negative curvature information, and confirm that curvature has even more chance of



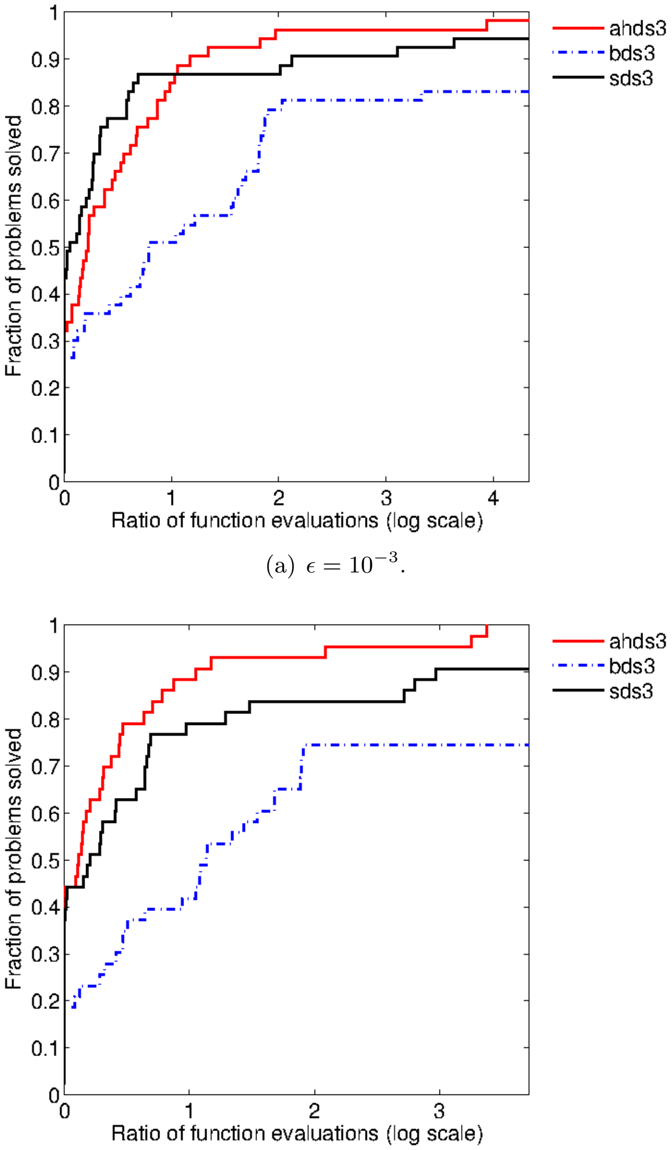
(a) $\epsilon = 10^{-3}$.

(b) $\epsilon = 10^{-6}$.

Figure 3: Second-, first- and weakly second-order direct-search methods, with polling choice 2 and a budget of $2000\,n$ evaluations.

being exploited by computing a Hessian approximation.

A final comment can be made by studying the relative performance among different instances of Algorithm 3.1 (see Figure 5). The method `ahds0` solves the most problems within the given budget, and also outperforms the other variants. Using symmetric positive spanning sets and building Hessian approximation with respect to orthonormal bases thus seems to allow exploiting more second-order information in general. Note again that the weak second-order properties of the polling set are particularly relevant, and likely to provide curvature information. As we have seen, completing the polling with a phase of Hessian approximation is an additional tool that



(a) $\epsilon = 10^{-3}$.

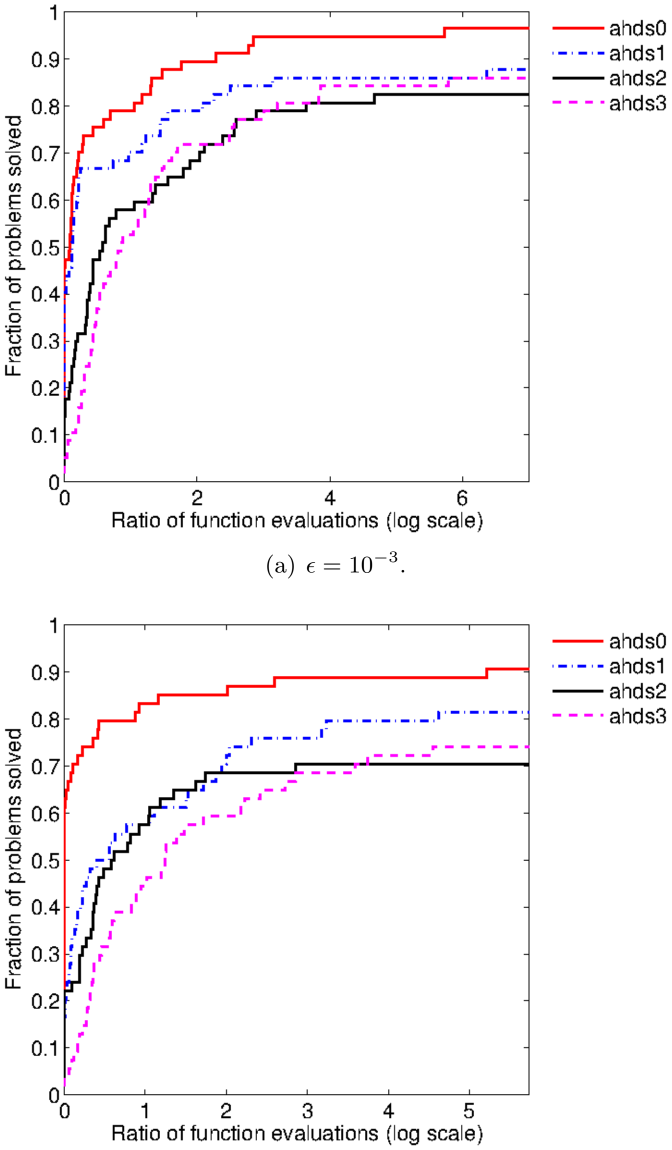
(b) $\epsilon = 10^{-6}$.

Figure 4: Second-, first- and weakly second-order direct-search methods, with polling choice 3 and a budget of $2000\,n$ evaluations.

appears to pay off in the long run.

6 Concluding remarks

As shown in this paper, weak second-order optimality can be established for a broad class of direct-search methods. Indeed, a measure depending on the directions considered by the method tends to be nonnegative; this measure is the best information one can use to approximate the minimum Hessian eigenvalues. For this measure to become a true second-order global



(a) $\epsilon = 10^{-3}$.

(b) $\epsilon = 10^{-6}$.

Figure 5: Comparison of the `ahds` methods, given a budget of $2000\,n$ evaluations.

convergence criterion, one needs to go further than a simple polling along vectors of a PSS. This was done by building a Hessian approximation, resulting in a provably second-order convergent direct-search instance. This technique also exhibits a worst-case complexity bound, that matches in order of the stationarity tolerances those of second-order convergent derivative-based methods. The dependence on the problem dimension is however worsened compared to the first-order case. This is due to the expense of constructing an approximated Hessian, which overwhelms the cost of polling using a PSS. Although pathological examples exist that illustrate the necessity of such approximations, it seems that the practical cost remains at a reasonable level, yet possibly higher than for the first-order case. This being said, if one can afford the expense in function evaluations, one benefits from exploiting negative curvature in practice, especially while searching for high accuracy solutions. Moreover, the sole introduction of weak second-order aspects (e.g. by symmetrizing the polling sets) can already improve the performance for problems where curvature is worth considering.

A natural follow-up to this study is to consider the strong second-order requirements of Section 3 in a probabilistic context. As in the first-order case [25], this could lead to an almost surely convergent method while improving both the worst-case complexity and the practical efficiency. Regarding the weak second-order results of Section 2, one might want to study their impact in the context of multilevel optimization methods, as it was done for the trust-region methods [26]. Indeed, the study of the curvature along some directions might be helpful in this regard, and direct-search methods have already been adapted to the multilevel framework [19]. Finally, a more difficult challenge would be to translate the features of curvilinear line searches from derivative-based optimization in a derivative-free environment; this could lead to the design of a general direct/line-search framework that would also be second-order convergent. Probabilistic assumptions may also play a significant role to reduce the cost of such algorithms.

References

[1] M. A. Abramson. Second-order behavior of pattern search. *SIAM J. Optim.*, 16:315–330, 2005.

[2] M. A. Abramson and C. Audet. Convergence of mesh adaptive direct search to second-order stationarity points. *SIAM J. Optim.*, 17:606–619, 2006.

[3] M. A. Abramson, L. Frimannslund, and T. Steihaug. A subclass of generating set search with convergence to second-order stationary points. *Optim. Methods Softw.*, 29:900–918, 2014.

[4] C. Audet and J. E. Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM J. Optim.*, 17:188–217, 2006.

[5] C. P. Avelino, J. M. Moguerza, A. Olivares, and F. J. Prieto. Combining and scaling descent and negative curvature directions. *Math. Program.*, 128:285–319, 2011.

[6] A. S. Bandeira, K. Scheinberg, and L. N. Vicente. Computation of sparse low degree interpolating polynomials and their application to derivative-free optimization. *Math. Program.*, 134:223–257, 2012.

[7] Á. Bűrmen, J. Olenšek, and T. Tuma. Mesh adaptive direct search with second directional derivative-based Hessian update. *Comput. Optim. Appl.*, 2015. To appear.

[8] E. G. Boman. *Infeasibility and Negative Curvature in Optimization.* PhD thesis, Stanford University, February 1999.

[9] 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.

[10] C. Cartis, N. I. M. Gould, and Ph. L. Toint. Complexity bounds for second-order optimality in unconstrained optimization. *J. Complexity*, 28:93–108, 2012.

[11] 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.

[12] A. R. Conn, K. Scheinberg, and L. N. Vicente. Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points. *SIAM J. Optim.*, 20:387–415, 2009.

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

[14] I. D. Coope and C. J. Price. A direct search conjugate directions algorithm for unconstrained minimization. *ANZIAM J.*, 42:C478–C498, 2000.

[15] C. Davis. Theory of positive linear dependence. *Amer. J. Math.*, 76:733–746, 1954.

[16] M. Dodangeh and L. N. Vicente. Worst case complexity of direct search under convexity. *Math. Program.*, 2015. To appear.

[17] M. Dodangeh, L. N. Vicente, and Z. Zhang. On the optimal order of worst case complexity of direct search. Technical Report 14-38, Dept. Mathematics, Univ. Coimbra, 2015.

[18] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Math. Program.*, 91:201–213, 2002.

[19] E. Frandi and A. Papini. Coordinate search algorithms in multilevel optimization. *Optim. Methods Softw.*, 29:1020–1041, 2014.

[20] L. Frimannslund and T. Steihaug. A generating set search method using curvature information. *Comput. Optim. Appl.*, 38:105–121, 2007.

[21] R. Garmanjani, D. Júdice, and L. N. Vicente. Trust-region methods without using derivatives: Worst case complexity and the non-smooth case. Technical Report 15-03, Dept. Mathematics, Univ. Coimbra, 2015.

[22] 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.

[23] N. I. M. Gould, S. Lucidi, M. Roma, and Ph. L. Toint. Exploiting negative curvature directions in linesearch methods for unconstrained optimization. *Optim. Methods Softw.*, 14:75–98, 2000.

[24] N. I. M. Gould, D. Orban, and Ph. L. Toint. CUTEst: a Constrained and Unconstrained Testing Environment with safe threads. *Comput. Optim. Appl.*, 60:545–557, 2015.

[25] S. Gratton, C. W. Royer, L. N. Vicente, and Z. Zhang. Direct search using probabilistic descent. *SIAM J. Optim.*, 2015. To appear.

[26] S. Gratton, A. Sartenaer, and Ph. L. Toint. Second-order convergence properties of trust-region methods using incomplete curvature information, with an application to multigrid optimization. *J. Comput. Math.*, 24:676–692, 2006.

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

[28] 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.

[29] J. Konečný and P. Richtárik. Simplified complexity analysis of simplified direct search. Technical Report ERGO 14-012, School of Mathematics, Edinburgh University, 2014.

[30] S. Lucidi, F. Rochetich, and M. Roma. Curvilinear stabilization techniques for truncated Newton methods in large scale unconstrained optimization. *SIAM J. Optim.*, 8:916–939, 1998.

[31] R. Mifflin. A superlinearly convergent algorithm for minimization without evaluating derivatives. *Math. Program.*, 9:100–117, 1975.

[32] J.J. Moré and D. C. Sorensen. On the use of directions of negative curvature in a modified Newton method. *Math. Program.*, 16:1–20, 1979.

[33] J.J. Moré and S. M. Wild. Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.*, 20:172–191, 2009.

[34] A. Olivares, J. M. Moguerza, and F. J. Prieto. Nonconvex optimization using negative curvature within a modified linesearch. *European J. Oper. Res.*, 189:706–722, 2008.

[35] V. Torczon. On the convergence of pattern search algorithms. *SIAM J. Optim.*, 7:1–25, 1997.

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