Chapter 7

Directional direct-search methods

Direct-search methods are derivative-free methods that sample the objective function at a finite number of points at each iteration and decide which actions to take next solely based on those function values and without any explicit or implicit derivative approximation or model building. In this book we divide the presentation of direct-search methods into two chapters. In the next chapter we cover direct-search methods based on simplices and operations over simplices, like reflections, expansions, or contractions. A classical example of a simplicial direct-search algorithm is the Nelder–Mead method.

In this chapter we address direct-search methods where sampling is guided by sets of directions with appropriate features. Of key importance in this chapter are the concepts of positive spanning sets and positive bases (see Section 2.1). The two classes of direct-search methods considered in this book (directional and simplicial) are related to each other. For instance, by recalling what we have seen in Section 2.5, one can easily construct maximal positive bases from any simplex of $n+1$ vertices. Reciprocally, given any positive basis, it is straightforward to identify simplices of $n+1$ vertices. Despite the intimacy of the two concepts (positive spanning and affine independency), the philosophy of the two classes of direct-search methods under consideration differ enough to justify different treatments.

The problem under consideration is the unconstrained optimization of a real-valued function, stated in (1.1). Extensions of directional direct-search methods for various types of derivative-free constrained optimization problems are summarized in Section 13.1.

7.1 The coordinate-search method

One of the simplest directional direct-search methods is called coordinate or compass search. This method makes use of the maximal positive basis $D_\oplus$:

$$D_\oplus = \begin{bmatrix} I & -I \end{bmatrix} = [\, e_1 \cdots e_n \; -e_1 \cdots -e_n \,]. \tag{7.1}$$

Let x_k be a current iterate and α_k a current value for the step size or mesh parameter. Coordinate search evaluates the function f at the points in the set

$$P_k = \{x_k + \alpha_k d : d \in D_\oplus\},$$

following some specified order, trying to find a point in P_k that decreases the objective function value. In the terminology of this chapter, we say that P_k is a set of poll points and $D_\oplus$ is a set of poll vectors or directions. This process of evaluating the objective function is called polling. We illustrate the poll process for coordinate search in Figure 7.1.

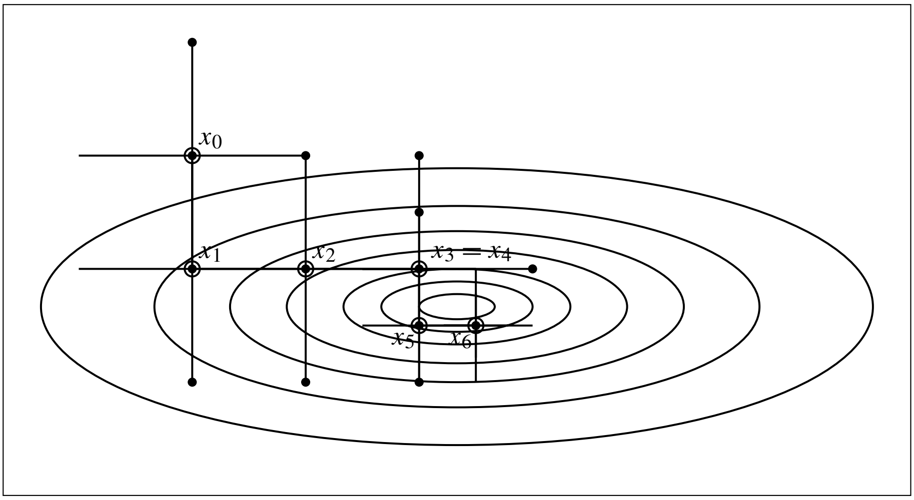

Figure 7.1. *First six iterations of coordinate search with opportunistic polling (following the order North/South/East/West). Function evaluations (a total of* 14*) occur at circles, but only the bigger circles are iterates. The ellipses depict the level sets of the function.*

Polling is successful when one of the points in P_k is better than the current iterate x_k in terms of the values of f. When that happens, the method defines a new iterate $x_{k+1} = x_k + \alpha_k d_k \in P_k$ such that $f(x_{k+1}) < f(x_k)$ (a *simple decrease* in the objective function). In such a successful case, one either leaves the parameter α_{k+1} unchanged or increases it (say by a factor of 2). If none of the points in P_k leads to a decrease in f, then the parameter α_k is reduced (say by a factor of $1/2$) and the next iteration polls at the same point ($x_{k+1} = x_k$). Polling can be opportunistic, moving to the first encountered better point, or complete, in which case all the poll points are evaluated and the best point is taken (if better than the current iterate). Complete polling is particularly attractive for running on a parallel environment.

Algorithm 7.1 (Coordinate-search method).

Initialization: Choose x_0 and $\alpha_0 > 0$.

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

1. **Poll step:** Order the poll set $P_k = \{x_k + \alpha_k d : d \in D_\oplus\}$. Start evaluating f at the poll points following the order determined. If a poll point $x_k + \alpha_k d_k$ is

found such that $f(x_k + \alpha_k d_k) < f(x_k)$, 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 $x_{k+1} = x_k$.

2. **Parameter update:** If the iteration was successful, set $\alpha_{k+1} = \alpha_k$ (or $\alpha_{k+1} = 2\alpha_k$). Otherwise, set $\alpha_{k+1} = \alpha_k/2$.

To illustrate what is coming later we also introduce for coordinate search the following set (called a mesh, or a grid):

$$M_k = \left\{ x_k + \alpha_k \left(\sum_{i=1}^{n} u_i e_i + \sum_{i=1}^{n} u_{n+i}(-e_i) \right) : u \in \mathbb{Z}_+^{|2n|} \right\}, \tag{7.2}$$

where $\mathbb{Z}_+$ is the set of nonnegative integers. An example of the mesh is illustrated in Figure 7.2. The mesh is merely conceptual. There is never an attempt in this class of methods to enumerate (computationally or not) points in the mesh.

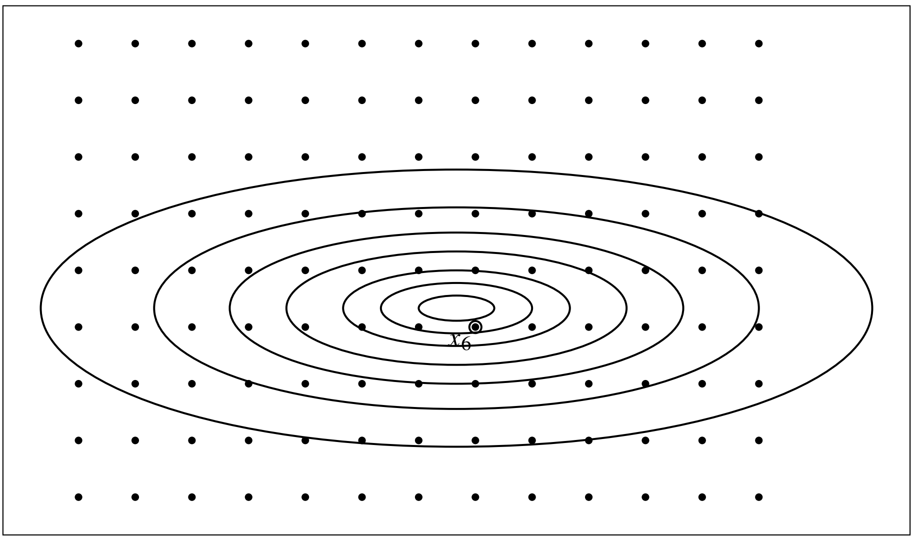

Figure 7.2. *The mesh in coordinate search at* x_6.

For matters of practical efficiency, it is useful to introduce some flexibility in the above coordinate-search framework. Such flexibility can be accommodated by the so-called search step,[8] which is optional and applied just before polling when formulated. Basically, the search step consists of evaluating the objective function at a finite number of points in M_k, trying to find a point $y \in M_k$ such that $f(y) < f(x_k)$. We illustrate this process in Figure 7.3. When the search step is successful so is the iteration (the poll step is skipped and a new iteration starts at $x_{k+1} = y$). The search step is totally optional, not only in the implementation of the method but also when proving its convergence properties. When the

[8] We should remark that coordinate search is often described in the literature without a search step.

search step is applied it has no interference in the convergence properties of the method since the points are required to be on the mesh M_k. In the next section, we will describe a class of directional direct-search methods that includes coordinate search as a special case.

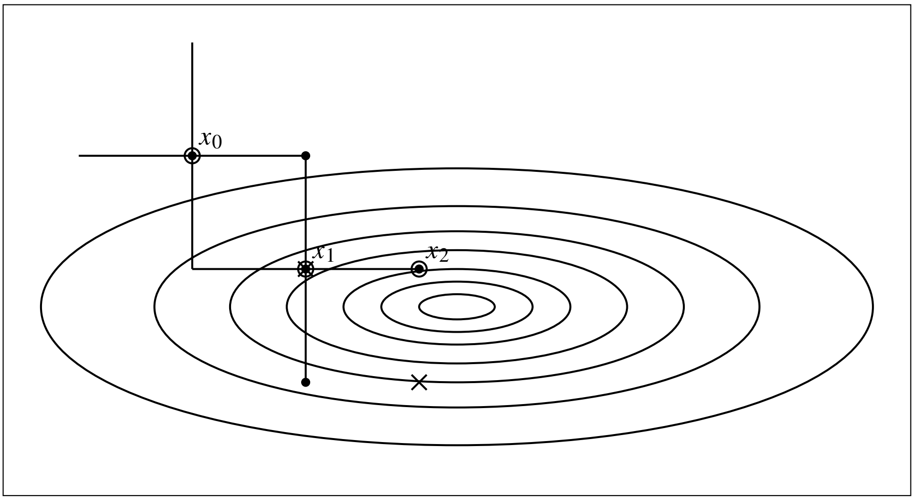

Figure 7.3. *Three iterations of coordinate search with a search step (consisting of trying the South-East point) and opportunistic polling (following the order North/South/East/West). Function evaluations occur at crosses (search step) and circles (a total of* 6*). Only the bigger circles are iterates. The ellipses depict the level sets of the function.*

7.2 A directional direct-search framework

We now present a class of globally convergent directional direct-search methods. Much of this presentation is based on the generalized pattern-search framework introduced by Audet and Dennis [18] and makes extensive use of the structure of an iteration organized around a search step and a poll step.

To start the presentation let us consider a current iterate x_k and a current value for the step size or mesh parameter α_k. The goal of iteration k of the direct-search methods presented here is to determine a new point x_{k+1} such that $f(x_{k+1}) < f(x_k)$.

The process of finding a new iterate x_{k+1} can be described in two phases (usually designated as the search step and the poll step).

The search step is optional and is not necessary for the convergence properties of the method. It consists of evaluating the objective function at a finite number of points. The choice of points is totally arbitrary as long as their number remains finite (later we will see that the points must be in a mesh M_k if only simple decrease is imposed, but we skip this issue to leave the presentation as conceptual as possible).[9] For example, the points

[9]It is obvious that descent in the search step must be controlled in some form. The reader can think of what a sequence of points of the form $x_k = 2 + 1/k$ does to the minimization of $f(x) = x^2$.

might be chosen according to specific application properties or following some heuristic algorithm. The search step can take advantage of the existence of surrogate models for f (see Chapter 12) to improve the efficiency of the direct-search method. The search step and the current iteration are declared successful if a new point x_{k+1} is found such that $f(x_{k+1}) < f(x_k)$.

The poll step is performed only if the search step has been unsuccessful. It consists of a local search around the current iterate, exploring a set of points defined by the step size parameter α_k and by a positive basis D_k:[10]

$$P_k = \{x_k + \alpha_k d : d \in D_k\}. \tag{7.3}$$

The points $x_k + \alpha_k d \in P_k$ are called the poll points and the vectors $d \in D_k$ the poll vectors or directions. Later we will see that the poll points must also lie in the mesh M_k if only simple decrease is imposed, but, again, we skip this issue to concentrate on the geometrical properties of these methods (which are related to those of other derivative-free methods).

The purpose of the poll step is to ensure a decrease of the objective function for a sufficiently small step size parameter α_k. As we saw in Section 2.2, as long as the objective function retains some differentiability properties and unless the current iterate is a stationary point, we know that the poll step must eventually be successful (after a finite number of reductions of the step size parameter). The key ingredient here is the fact that there is at least one descent direction in each positive basis D_k.

The poll step and the current iteration are declared successful if a new point $x_{k+1} \in P_k$ is found such that $f(x_{k+1}) < f(x_k)$. If the poll step fails to produce a point in P_k where the objective function is lower than $f(x_k)$, then both the poll step and the iteration are declared unsuccessful. In these circumstances the step size parameter α_k is typically decreased.

The step size parameter is kept unchanged (or possibly increased) if the iteration is successful (which happens if either in the search step or in the poll step a new iterate is found yielding objective function decrease).

In this class of directional direct-search methods one can consider multiple positive bases and still be able to guarantee global convergence to stationary points. When new iterates are accepted based on *simple decrease* of the objective function (as we have just described), the number of positive bases is required to be finite. As we will point out later, this requirement can be relaxed if one imposes a *sufficient decrease* condition to accept new iterates. Still, in such a case, one can use only an infinite number of positive bases for which the cosine measure is uniformly bounded away from zero.

The class of directional direct-search methods analyzed in this book is described in Algorithm 7.2. Our description follows the one given in [18] for the generalized pattern search, by considering search and poll steps separately. We do not specify for the moment the set $\mathcal{D}$ of positive bases used in the algorithm. Polling is opportunistic, moving to the first encountered better point. The poll vectors (or points) are ordered according to some criterion in the poll step. In many papers and implementations this ordering is the one in which they were originally stored, and it is never changed during the course of the iterations. Consequently, our presentation of directional direct search considers that the poll directions are ordered in some given form before (opportunistic) polling starts. From a theoretical point of view, this ordering does not matter and could change at every

[10] The application of this class of direct-search methods and its convergence properties is valid both for positive spanning sets and positive bases (satisfying some properties mentioned later).

iteration. Efficient procedures to order the poll directions include ordering according to the angle proximity to a negative simplex gradient, random ordering, and ordering following the original order but avoiding restarts at new poll iterations (and combinations of these strategies).

Algorithm 7.2 (Directional direct-search method).

Initialization: Choose x_0, $\alpha_0 > 0$, $0 < \beta_1 \leq \beta_2 < 1$, and $\gamma \geq 1$. Let $\mathcal{D}$ be a set of positive bases.

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

1. **Search step:** Try to compute a point with $f(x) < f(x_k)$ 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 basis D_k from the set $\mathcal{D}$. Order the poll set $P_k = \{x_k + \alpha_k d : d \in D_k\}$. Start evaluating f at the poll points following the chosen order. If a poll point $x_k + \alpha_k d_k$ is found such that $f(x_k + \alpha_k d_k) < f(x_k)$, 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: $\alpha_{k+1} \in [\beta_1\alpha_k, \beta_2\alpha_k]$.

The poll step makes at most $|D_k|$ (where $|D_k| \geq n+1$) function evaluations and exactly that many at all unsuccessful iterations.

The natural stopping criterion in directional direct search is to terminate the run when $\alpha_k < \alpha_{tol}$, for a chosen tolerance $\alpha_{tol} > 0$ (for instance, $\alpha_{tol} = 10^{-5}$).

7.3 Global convergence in the continuously differentiable case

First, we point out that this class of directional direct-search methods is traditionally analyzed under the assumption that all iterates lie in a compact set. Given that the sequence of iterates $\{x_k\}$ is such that $\{f(x_k)\}$ is monotonically decreasing, a convenient way of imposing this assumption is to assume that the level set $L(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is compact.

Assumption 7.1. *The level set* $L(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ *is compact.*

We are interested in analyzing the global convergence properties of these methods, meaning convergence to stationary points from arbitrary starting points. In direct search we will deal only with first-order stationary points.

Behavior of the step size parameter

The global convergence analysis for these direct-search methods relies on proving first that there exists a subsequence of step size parameters converging to zero. For this purpose we must be able to ensure the following assumption.

Assumption 7.2. *If there exists an $\alpha > 0$ such that $\alpha_k > \alpha$, for all k, then the algorithm visits only a finite number of points.*

In Sections 7.5 and 7.7, we will discuss how this assumption can be ensured in practical implementations of direct search. We will see in Section 7.5 that when the number of positive bases is finite, some integral/rational structure on the construction of these bases and on the update of the step size parameter will suffice for this purpose. When any number of bases is allowed, then something else is required to achieve Assumption 7.2 (namely a sufficient decrease condition to accept new iterates, as we will prove in Section 7.7).

Based on Assumption 7.2 one can prove that the step size parameter tends to zero.

Theorem 7.1. *Let Assumption* 7.2 *hold. Then the sequence of step size parameters satisfies*

$$\liminf_{k \longrightarrow +\infty} \alpha_k = 0.$$

Proof. Let us assume, by contradiction, that there exists an $\alpha > 0$ such that $\alpha_k > \alpha$ for all k. Then one knows from Assumption 7.2 that the number of points visited by the algorithm is finite. On the other hand, the algorithm moves to a different point only when a decrease in the objective function is detected. By putting these last two arguments together, we arrive at the conclusion that there must exist an iteration $\bar{k}$ such that $x_k = x_{\bar{k}}$ for all $k \geq \bar{k}$. From the way α_k is updated in unsuccessful iterations, it follows that $\lim_{k \longrightarrow +\infty} \alpha_k = 0$, which contradicts what we have assumed at the beginning of the proof. $\square$

The following corollary follows from Assumption 7.1.

Corollary 7.2. *Let Assumptions* 7.1 *and* 7.2 *hold. There exist a point x_* and a subsequence $\{k_i\}$ of unsuccessful iterates for which*

$$\lim_{i \longrightarrow +\infty} \alpha_{k_i} = 0 \qquad \textit{and} \qquad \lim_{i \longrightarrow +\infty} x_{k_i} = x_*. \tag{7.4}$$

Proof. Theorem 7.1 states the existence of an infinite subsequence of the iterates driving the step size parameter to zero. As a result, there must exist an infinite subsequence of iterations corresponding to unsuccessful poll steps, since the step size parameter is reduced only at such iterations. Let K_u^1 denote the index subsequence of all unsuccessful poll steps.

It follows also from the scheme that updates the step size parameter and from the above observations that there must exist a subsequence $K_u^2 \subset K_u^1$ such that $\alpha_{k+1} \to 0$ for $k \in K_u^2$. Since, $\alpha_k \leq (1/\beta_1)\alpha_{k+1}$ for $k \in K_u^2$, we obtain $\alpha_k \to 0$ for $k \in K_u^2$.

Since $\{x_k\}_{K_u^2}$ is bounded, it contains a convergent subsequence $\{x_k\}_{K_u^3}$. Let $x_* = \lim_{k \in K_u^3} x_k$. Since $K_u^3 \subset K_u^2$, it also holds that $\lim_{k \in K_u^3} \alpha_k = 0$, and the proof is completed by setting $\{k_i\} = K_u^3$. $\square$

The rest of the global convergence analysis of directional direct search relies on the geometrical properties of positive bases and on differentiability properties of f either on the entire level set $L(x_0)$ or at the limit point x_* identified in Corollary 7.2. We will consider the most relevant cases next.

Arbitrary set of positive bases

Directional direct-search methods can make use of an infinite number of positive bases as long as they do not become degenerate, namely, as long as their cosine measures are uniformly bounded away from zero. We frame this assumption next. It is also required to bound the size of all vectors in all positive bases used.

Assumption 7.3. *Let $\xi_1, \xi_2 > 0$ be some fixed positive constants. The positive bases D_k used in the algorithm are chosen from the set*

$$\mathcal{D} = \left\{ \bar{D} \text{ positive basis} : \ \mathrm{cm}(\bar{D}) > \xi_1, \ \|\bar{d}\| \le \xi_2, \ \bar{d} \in \bar{D} \right\}.$$

In addition, when using an infinite number of bases we require that the gradient of f is Lipschitz continuous on the level set $L(x_0)$.

Assumption 7.4. *The gradient ∇f is Lipschitz continuous in an open set containing $L(x_0)$ (with Lipschitz constant $\nu > 0$).*

We will see later that this assumption is not necessary if one uses a finite number of positive bases. By assuming the Lipschitz continuity we can use the result of Theorem 2.8 and, thus, relate the global convergence of this type of direct search to the global convergence of the derivative-free methods based on line searches or trust regions.

Under this assumption we arrive at our first global convergence result for the class of direct-search methods considered.

Theorem 7.3. *Let Assumptions 7.1, 7.2, 7.3, and 7.4 hold. Then*

$$\liminf_{k \longrightarrow +\infty} \|\nabla f(x_k)\| = 0,$$

and the sequence of iterates $\{x_k\}$ has a limit point x_* (given in Corollary 7.2) for which

$$\nabla f(x_*) = 0.$$

Proof. Corollary 7.2 showed the existence of a subsequence $\{k_i\}$ of unsuccessful iterations (or unsuccessful poll steps) for which (7.4) is true.

From Theorem 2.8 (which can be applied at unsuccessful poll steps and for k_i sufficiently large), we get that

$$\|\nabla f(x_{k_i})\| \le \frac{\nu}{2} \mathrm{cm}(D_{k_i})^{-1} \max_{d \in D_{k_i}} \|d\| \alpha_{k_i}. \tag{7.5}$$

As a result of Assumption 7.3, we obtain

$$\|\nabla f(x_{k_i})\| \le \frac{\nu \xi_2}{2 \xi_1} \alpha_{k_i}.$$

Thus, we conclude that

$$\lim_{i \longrightarrow +\infty} \|\nabla f(x_{k_i})\| = 0,$$

which shows the first part of the theorem. Since ∇f is continuous, $x_* = \lim_{i \longrightarrow +\infty} x_{k_i}$ is such that $\nabla f(x_*) = 0$. □

One could also obtain the result of Theorem 7.3 by assuming the Lipschitz continuity of ∇f near x_* (meaning in a ball containing x_*), where x_* is the point identified in Corollary 7.2. Note that to obtain this result it is not enough to assume that ∇f is Lipschitz continuous near all the stationary points of f since, as we will see in Section 7.4, the point x_* in Corollary 7.2 might not be a stationary point if smoothness is lacking.

The step size parameter provides a natural stopping criterion for directional direct-search methods, since not only is there a subsequence of step size parameters converging to zero (Theorem 7.1), but one also has (in the continuously differentiable case) that $\nabla f(x_k) = \mathcal{O}(\alpha_k)$ after an unsuccessful iteration (see, again, Theorem 2.8 or (7.5) above). In general, α_k seems to be a reasonable measure of stationarity (even in the nonsmooth case). Dolan, Lewis, and Torczon [80] studied this issue in detail.[11] They reported results indicating that in practice one also observes that $\alpha_k = \mathcal{O}(\nabla f(x_k))$, a hypothesis confirmed by our numerical experience.

Finite set of positive bases

We will now prove global convergence under the assumption that the number of positive bases is finite, using an argument different from the proof of Theorem 7.3.

The proof of Theorem 7.3 goes through when the set of all positive bases is infinite, provided $\mathrm{cm}(D_k)^{-1}$ is uniformly bounded. The argument used in Theorem 7.4 below, however, is heavily dependent on a finite number of different D_k's.

Both proofs have their own advantages. The proof of Theorem 7.4 is easily generalized to the nonsmooth case as we will see next. The proof of Theorem 7.3 not only is more of the style of the ones applied to analyze other methods in this book but also allows the use of an infinite number of positive bases (provided their cosine measure is uniformly bounded away from zero).

Assumption 7.5. *The set $\mathcal{D}$ of positive bases used by the algorithm is finite.*

In this case, it is enough to require the continuity of the gradient of f. The following assumption is the counterpart, for continuous differentiability of the objective function, of Assumption 7.4.

Assumption 7.6. *The function f is continuously differentiable in an open set containing $L(x_0)$.*

[11] In their paper it is also shown that *pattern-search methods* (directional direct-search methods based on integer lattices, as explained in Section 7.5) produce sequences of iterates for which the subsequence of unsuccessful iterates converges r-linearly to x_* (in the case where α_k is not increased at successful iterations after some finite iteration).

Now we prove, for continuously differentiable functions f, the same result as in Theorem 7.3 but following a different proof.

Theorem 7.4. *Let Assumptions* 7.1, 7.2, 7.5, *and* 7.6 *hold. Then the sequence of iterates* $\{x_k\}$ *has a limit point* x_* *(given in Corollary* 7.2*) for which*

$$\nabla f(x_*) = 0.$$

Proof. Recall the definitions of the poll set P_k and of an unsuccessful iteration (which includes an unsuccessful poll step). The following is true for any unsuccessful iteration k (such that f is continuously differentiable at x_k):

$$\begin{aligned} f(x_k) \le \min_{x \in P_k} f(x) &= \min_{x \in \{x_k + \alpha_k d : d \in D_k\}} f(x) \\ &= \min_{d \in D_k} f(x_k + \alpha_k d) \\ &= \min_{d \in D_k} f(x_k) + \nabla f(x_k + t_{k,d} \alpha_k d)^\top (\alpha_k d) \\ &= f(x_k) + \alpha_k \min_{d \in D_k} \nabla f(x_k + t_{k,d} \alpha_k d)^\top d, \end{aligned}$$

where $t_{k,d} \in (0,1)$ depends on k and d, and consequently

$$0 \le \min_{d \in D_k} \nabla f(x_k + t_{k,d} \alpha_k d)^\top d.$$

Corollary 7.2 showed the existence of a subsequence of unsuccessful iterations $\{k_i\}$ for which (7.4) is true. The above inequality is true for this subsequence $\{k_i\}$. Since the number of positive bases is finite, there exists at least one $D_* \subset \mathcal{D}$ that is used an infinite number of times in $\{k_i\}$. Thus,

$$0 \le \min_{d \in D_*} \nabla f(x_*)^\top d. \tag{7.6}$$

Inequality (7.6) and the property of the spanning sets given in Theorem 2.3(iv) necessarily imply $\nabla f(x_*) = 0$. ◻

One could also obtain the result of Theorem 7.4 by assuming the continuity of ∇f near x_* (meaning in a ball containing x_*), where x_* is the point identified in Corollary 7.2.

7.4 Global convergence in the nonsmooth case

In the nonsmooth case one cannot expect directional direct search to globally converge to stationarity. In Figure 7.4 we depict the contours of the two-dimensional real function:

$$f(x) = \frac{1}{2} \max\left\{ \|x - c_1\|^2, \|x - c_2\|^2 \right\}, \tag{7.7}$$

where $c_1 = (1,-1)$ and $c_2 = -c_1$. This function, introduced in [145], is a variant of the Dennis–Woods function [78]. The function is continuous and strictly convex everywhere, but its gradient is discontinuous along the line $x_1 = x_2$. The function has a strict minimizer at $(0,0)$.

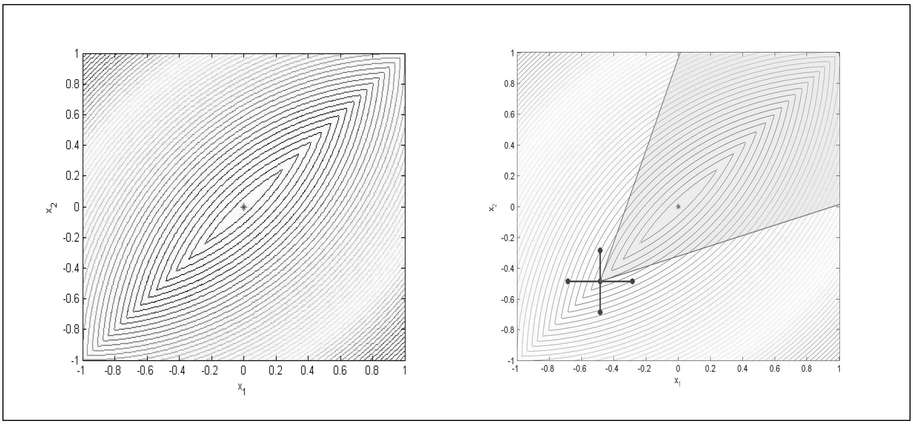

Figure 7.4. *Contours of the Dennis–Woods-type function* (7.7) *for* $c_1 = (1,-1)$ *and* $c_2 = -c_1$. *The cone of descent directions at the poll center is shaded.*

It has also been pointed out in [145] that coordinate search can fail to converge on this function. The reader can immediately see that at any point of the form (a,a), with $a \neq 0$, coordinate search generates an infinite number of unsuccessful iterations without any progress. In fact, none of the elements of $D_\oplus = [e_1\ e_2\ -e_1\ -e_2]$ is a descent direction (see Figure 7.4). The descent directions of f at (a,a), with $a \neq 0$, are marked in the shaded region of the picture. Our numerical experience has not led to the observation (reported in [145]) that coordinate search frequently tends to converge to points of this form where, then, stagnation easily occurs. In fact, we found that stagnation occurs only when the starting points are too close to points on this line, as illustrated in Figure 7.5.

It is possible to prove, though, that directional direct search can generate a sequence of iterates under Assumptions 7.1, 7.2, and 7.5 which has a limit point where directional derivatives are nonnegative for all directions in a positive basis. Such a statement may not be a certificate of any type of stationarity (necessary conditions for optimality), as the example above would immediately show.

Let us consider the point x_* identified in Corollary 7.2. We will assume that f is Lipschitz continuous near x_* (meaning in a neighborhood of x_*), so that the generalized directional derivative (in the Clarke sense [54]) can assume the form

$$f^\circ(x;d) \;=\; \limsup_{y\to x,\, t\downarrow 0} \frac{f(y+td)-f(y)}{t}$$

for all directions $d \in \mathbb{R}^n$. Since f is Lipschitz continuous near x_*, this limit is well defined, and so is the generalized subdifferential (or subgradient)

$$\partial f(x_*) \;=\; \{s \in \mathbb{R}^n :\ f^\circ(x_*;v) \geq v^\top s \quad \forall v \in \mathbb{R}^n\}.$$

Moreover,

$$f^\circ(x_*;d) \;=\; \max\{d^\top s :\ s \in \partial f(x_*)\}.$$

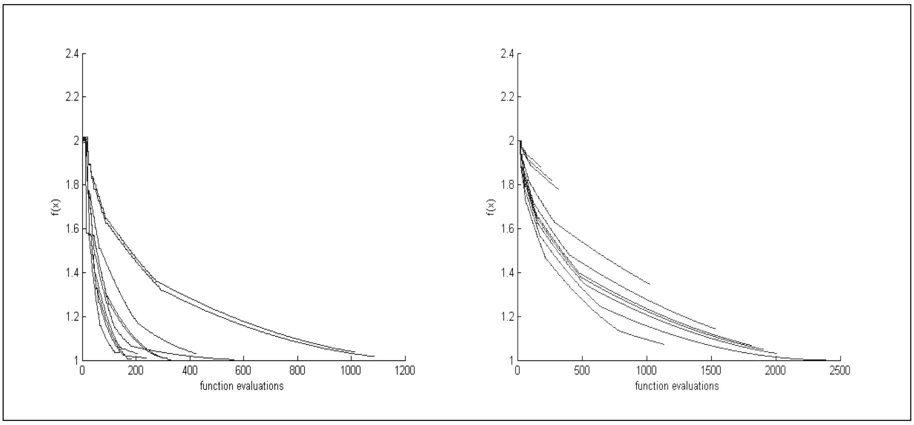

Figure 7.5. *Application of the coordinate-search method to the Dennis–Woods function* (7.7) *starting around the point* $x_0 = (1,1)$. *The plots on the left (resp., right) correspond to* 10 *starting points randomly generated in a box of* ℓ_∞ *radius* 10^{-2} *(resp.,* 10^{-3}*) around the point* $(1,1)$.

Assumption 7.7. *Let* x_* *be the point identified in Corollary* 7.2*, and let the function* f *be Lipschitz continuous near* x_*.

Theorem 7.5. *Let Assumptions* 7.1*,* 7.2*,* 7.5*, and* 7.7 *hold. Then the sequence of iterates* $\{x_k\}$ *has a limit point* x_* *(given in Corollary* 7.2*) for which*

$$f^\circ(x_*;d) \geq 0 \quad \forall d \in D_*,$$

where D_* *is one of the positive bases in* $\mathcal{D}$.

Proof. Corollary 7.2 showed the existence of a subsequence of unsuccessful iterations $\{k_i\}$ for which (7.4) is true. Since the number of positive bases used is finite, there exists one positive basis $D_* \subset \mathcal{D}$ for which

$$f(x_{k_i} + \alpha_{k_i} d) \geq f(x_{k_i})$$

for all $d \in D_*$ (and all i sufficiently large).

From the definition of the generalized directional derivative, we get, for all $d \in D_*$, that

$$f^\circ(x_*;d) = \limsup_{y \to x_*, t \downarrow 0} \frac{f(y+td) - f(y)}{t} \geq \limsup_{k \in \{k_i\}} \frac{f(x_k + \alpha_k d) - f(x_k)}{\alpha_k} \geq 0.$$

The proof is completed. $\square$

If, in addition to Assumption 7.7, the function f is regular at x_* (meaning that the directional derivative $f'(x_*;v)$ exists and coincides with the generalized directional derivative $f^\circ(x_*;v)$ for all $v \in \mathbb{R}^n$; see [54]), then the result of Theorem 7.5 becomes

$$f'(x_*;d) \geq 0 \quad \forall d \in D_*,$$

where D_* is one of the positive bases in $\mathcal{D}$. Neither the result of Theorem 7.5 nor this particularization for regular functions implies stationarity at x_*, as expected, since $D_* \neq \mathbb{R}^n$ and as the example above demonstrates.

Further, if the function f is, so called, strictly differentiable at x_* (which is equivalent to saying that f is Lipschitz continuous near x_* and there exists a vector $w = \nabla f(x_*)$—"the gradient"—such that

$$f^\circ(x_*; v) \;=\; w^\top v$$

for all $v \in \mathbb{R}^n$; see [54]), then the result of Theorem 7.5 becomes $\nabla f(x_*)^\top d \geq 0$ for all $d \in D_*$. Then the property about positive spanning sets given in Theorem 2.3(iv) implies that $\nabla f(x_*) = 0$, which is similar to what we obtained in the continuously differentiable case (Theorem 7.4).

Note that coordinate search can still fail to converge if strict differentiability is not assumed at the point x_* of Corollary 7.2. In [145], the authors provided the example

$$\hat{f}(x) \;=\; \left(1 - e^{-10^3\|x\|^2}\right) f(x),$$

where $f(x)$ is the modified Dennis–Woods function defined in (7.7). The function $\hat{f}(x)$ is just slightly different from $f(x)$ but is now strictly differentiable at the minimizer $(0,0)$ (but still not strictly differentiable at (a,a) with $a \neq 0$). However, the same problem can occur as before: coordinate search might find a point of the form (a,a), with $a \neq 0$, and stop since none of the directions in $D_\oplus$ provides descent at the poll steps, no matter how small the step size parameter is.

7.5 Simple decrease with integer lattices

We start by characterizing the directions in $\mathcal{D}$ used for polling. We assume $\mathcal{D}$ is finite and $\mathcal{D} = D$. As pointed out before in this book, it is convenient to regard D_k as an $n \times |D_k|$ matrix whose columns are the vectors in D_k, and, similarly, we regard the finite set D as an $n \times |D|$ matrix whose columns are the vectors in D.

Assumption 7.8. *The set $\mathcal{D} = D$ of positive bases used by the algorithm is finite. In addition, the columns of D are of the form $G\bar{z}_j$, $j = 1, \ldots, |D|$, where $G \in \mathbb{R}^{n\times n}$ is a nonsingular matrix and each $\bar{z}_j$ is a vector in $\mathbb{Z}^n$.*

Let $\bar{Z}$ denote the matrix whose columns are $\bar{z}_j$, $j = 1, \ldots, |D|$. We can therefore write $D = G\bar{Z}$. The matrix G is called a mesh or pattern generator.

In this section we will impose that all points generated by the algorithm lie on a mesh M_k defined by all possible nonnegative integer combinations of vectors in D:

$$M_k \;=\; \left\{ x_k + \alpha_k D u \,:\; u \in \mathbb{Z}_+^{|D|} \right\}, \tag{7.8}$$

where $\mathbb{Z}_+$ is the set of nonnegative integers. The mesh M_k is centered at the current iterate x_k, and its discretization size is defined by the step size or mesh size parameter α_k. It is easy to see that the mesh (7.2) introduced for coordinate search is a particular case of M_k when $D = D_\oplus$.

Note that the range of u in the definition of M_k allows the choice of the vectors in the canonical basis of $\mathbb{R}^{|D|}$. Thus, all points of the form $x_k + \alpha_k d$, $d \in D_k$, are in M_k for any

$D_k \subset \mathcal{D}$. It is clear that $P_k \subset M_k$, and thus we need only impose the following condition on the search step.

Assumption 7.9. *The search step in Algorithm* 7.2 *evaluates only points in M_k defined by* (7.8) *for all iterations k.*

A standard way to globalize directional direct-search-type methods is to force the iterates to lie on integer lattices. This intention is accomplished by imposing Assumptions 7.8 and 7.9 and the following additional assumption.

Assumption 7.10. *The step size parameter is updated as follows: Choose a rational number $\tau > 1$, a nonnegative integer $m^+ \geq 0$, and a negative integer $m^- \leq -1$. If the iteration is successful, the step size parameter is maintained or increased by taking $\alpha_{k+1} = \tau^{m_k^+}\alpha_k$, with $m_k^+ \in \{0,\ldots,m^+\}$. Otherwise, the step size parameter is decreased by setting $\alpha_{k+1} = \tau^{m_k^-}\alpha_k$, with $m_k^- \in \{m^-,\ldots,-1\}$.*

Note that these rules respect those of Algorithm 7.2 by setting $\beta_1 = \tau^{m^-}$, $\beta_2 = \tau^{-1}$, and $\gamma = \tau^{m^+}$.

First, we prove an auxiliary result from [18] which is interesting in its own right. This result states that the minimum distance between any two distinct points in the mesh M_k is bounded from below by a multiple of the mesh parameter α_k.

Lemma 7.6. *Let Assumption* 7.8 *hold. For any integer $k \geq 0$, one has that*

$$\min_{\substack{y,w \in M_k \\ y \neq w}} \|y - w\| \;\geq\; \frac{\alpha_k}{\|G^{-1}\|}.$$

Proof. Let $y = x_k + \alpha_k D u_y$ and $w = x_k + \alpha_k D u_w$ be two distinct points in M_k, where $u_y, u_w \in \mathbb{Z}_+^{|D|}$ (with $u_y \neq u_w$). Then

$$\begin{aligned} 0 \;\neq\; \|y - w\| &= \alpha_k \|D(u_y - u_w)\| \\ &= \alpha_k \|G\bar{Z}(u_y - u_w)\| \\ &\geq \alpha_k \frac{\|\bar{Z}(u_y - u_w)\|}{\|G^{-1}\|} \\ &\geq \frac{\alpha_k}{\|G^{-1}\|}. \end{aligned}$$

The last inequality is due to the fact that the norm of a vector of integers not identically zero, like $\bar{Z}(u_y - u_w)$, is never smaller than one. $\square$

It is important to remark that this result is obtained under Assumption 7.8, where the integrality requirement on the generation of the meshes plays a key role. For instance, all the positive integer combinations of directions in $\{-1, +\pi\}$ are dense in the real line, which does not happen with $\{-1, +1\}$. What is important is to guarantee a separation bounded away from zero for a fixed value of the step size parameter, and integrality is a convenient way of guaranteeing that separation.

Now we show that the sequence of step size or mesh parameters is bounded.

Lemma 7.7. *Let Assumptions* 7.1, 7.8, 7.9, *and* 7.10 *hold. There exists a positive integer* r^+ *such that* $\alpha_k \le \alpha_0 \tau^{r^+}$ *for any* $k \in \mathbb{N}_0$.

Proof. Since $L(x_0)$ is compact, one can consider

$$\theta \;=\; \max_{y,w \in L(x_0)} \|y - w\|.$$

Now suppose that $\alpha_k > \theta \|G^{-1}\|$ for some $k \in \mathbb{N}_0$. Then Lemma 7.6, with $w = x_k$, would show us that any $y \in M_k$, different from x_k, would not belong to $L(x_0)$. Thus, if $\alpha_k > \theta \|G^{-1}\|$, then iteration k would not be successful and $x_{k+1} = x_k$.

The step size parameter could pass the bound $\theta \|G^{-1}\|$ when it is lower than it. When it does, it must be at a successful iteration, and it cannot go above $\tau^{m^+} \theta \|G^{-1}\|$, where m^+ is the upper bound on m_k^+. The sequence $\{\alpha_k\}$ must, therefore, be bounded by $\tau^{m^+} \theta \|G^{-1}\|$. Letting r^+ be an integer such that $\tau^{m^+} \theta \|G^{-1}\| \le \alpha_0 \tau^{r^+}$ completes the proof. ∎

Since α_{k+1} is obtained by multiplying α_k by an integer power of τ, we can write, for any $k \in \mathbb{N}_0$, that

$$\alpha_k \;=\; \alpha_0 \tau^{r_k} \tag{7.9}$$

for some r_k in $\mathbb{Z}$. We now show that under the assumptions imposed in this section, Algorithm 7.2 meets the assumption used before for global convergence.

Theorem 7.8. *Let Assumptions* 7.1, 7.8, 7.9, *and* 7.10 *hold. If there exists an* $\alpha > 0$ *such that* $\alpha_k > \alpha$*, for all k, then the algorithm visits only a finite number of points. (In other words, Assumption* 7.2 *is satisfied.)*

Proof. The step size parameter is of the form (7.9), and hence to show the result we define a negative integer r^- such that $0 < \alpha_0 \tau^{r^-} \le \alpha_k$ for all $k \in \mathbb{N}_0$. Thus, from Lemma 7.7, we conclude that r_k must take integer values in the set $\{r^-, r^- + 1, \ldots, r^+\}$ for all $k \in \mathbb{N}_0$.

One knows that x_{k+1} can be written, for successful iterations k, as $x_k + \alpha_k D u_k$ for some $u_k \in \mathbb{Z}_+^{|D|}$. In unsuccessful iterations, $x_{k+1} = x_k$ and $u_k = 0$. Replacing α_k by $\alpha_0 \tau^{r_k}$, we get, for any integer $\ell \ge 1$,

$$\begin{aligned} x_\ell &= x_0 + \sum_{k=0}^{\ell-1} \alpha_k D u_k \\ &= x_0 + \alpha_0 G \sum_{k=0}^{\ell-1} \tau^{r_k} \bar{Z} u_k \\ &= x_0 + \frac{p^{r^-}}{q^{r^+}} \alpha_0 G \sum_{k=0}^{\ell-1} p^{r_k - r^-} q^{r^+ - r_k} \bar{Z} u_k, \end{aligned}$$

where p and q are positive integer numbers satisfying $\tau = p/q$. Since

$$\sum_{k=0}^{\ell-1} p^{r_k - r^-} q^{r^+ - r_k} \bar{Z} u_k$$

is a vector of integers for all $\ell \in \mathbb{N}$, we have just proved that the sequence of iterates $\{x_k\}$ lies in a set of the form (an integer lattice)

$$\mathcal{L} = \{x_0 + G_0 z \; : \; z \in \mathbb{Z}^n\},$$

where

$$G_0 = \frac{p^{r^-}}{q^{r^+}} \alpha_0 G$$

is a nonsingular $n \times n$ matrix. Now note that the intersection of $\mathcal{L}$ with the compact $L(x_0)$ is necessarily a finite set, which shows that the algorithm must visit only a finite number of points. $\square$

It is important to stress that no properties of f are specifically required in Theorem 7.8.

The rest of this section focuses on particular cases of the direct-search framework presented and on some extensions which preserve the asymptotic behavior of the step size parameter. A reader not acquainted with the convergence properties of these directional direct-search methods might postpone the rest of this section to a future reading.

Tightness of the integrality and rationality requirements

Assumptions 7.8 and 7.10 are necessary for Theorems 7.1 and 7.8 to hold, when a finite number of positive bases is used and only simple decrease imposed.

It is possible to show that the requirement of integrality stated in Assumption 7.8 for the positive bases cannot be lifted. An example constructed by Audet [12] shows an instance of Algorithm 7.2 which does not meet the integrality requirement of Assumption 7.8 for the positive bases and for which the step size parameter α_k is uniformly bounded away from zero when applied to a particular smooth function f.

Audet [12] also proved that the requirement of rationality on τ is tight. He provided an instance of Algorithm 7.2 for an irrational choice of τ, which for a given function f generates step size parameters α_k uniformly bounded away from zero. We point out that the function f used in this counterexample is discontinuous, which is acceptable under the assumptions of this section.

When $\tau > 1$ is an integer (and not just rational) the analysis above can be further simplified. In fact, one can easily see that $q = 1$ in the proof of Theorem 7.8 and the upper bound r^+ on r_k is no longer necessary.

Other definitions for the mesh

Instead of as in (7.8), the mesh M_k could be defined more generally as

$$M_k = \{x_k + \alpha_k D u : \; u \in \mathcal{Z}\}$$

as long as the set $\mathcal{Z} \subset \mathbb{Z}^{|D|}$ contains all the vectors of the canonical basis of $\mathbb{R}^{|D|}$ (so that $P_k \subset M_k$). Another possible generalization is sketched in the exercises.

For instance one could set

$$M_k = \{x_k + \alpha_k(jd) : \ d \in D, \ j \in \mathbb{Z}_+\},$$

which would amount to considering only mesh points along the vectors $d \in D$. Figure 7.6 displays two meshes M_k when $n = 2$, for the cases where D contains one and three maximal positive bases.

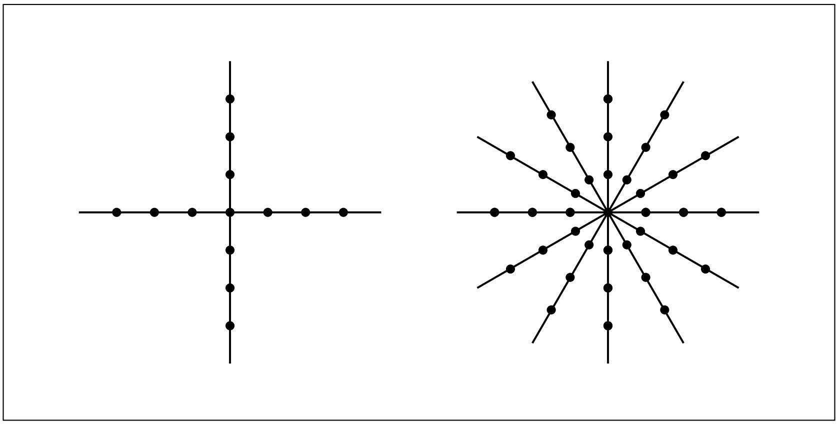

Figure 7.6. *Two pointed meshes, when D has one maximal positive basis (left) and three maximal positive bases (right).*

Complete polling and asymptotic results

Another example is provided by Audet [12], which shows that the application of an instance of Algorithm 7.2 under Assumption 7.8 to a continuously differentiable function can generate an infinite number of limit points, one of them not being stationary. Thus, the result

$$\liminf_{k \longrightarrow +\infty} \|\nabla f(x_k)\| = 0$$

cannot be extended to

$$\lim_{k \longrightarrow +\infty} \|\nabla f(x_k)\| = 0, \tag{7.10}$$

without further assumptions or modifications to the direct-search schemes in Algorithm 7.2. Such an observation is consistent with a similar one in trust-region algorithms for unconstrained nonlinear optimization, pointed out by Yuan [236]. This author constructed an example where a trust-region method based on simple decrease to accept new points (rather than a sufficient decrease condition—see Chapter 10) generates a sequence of iterates that does not satisfy (7.10) either.

Conditions under which it is possible to obtain (7.10) have been analyzed by Torczon [217] (see also [145]). The modifications in the directional direct-search framework are essentially two.

First, it is required that

$$\lim_{k \longrightarrow +\infty} \alpha_k = 0.$$

From Theorem 7.1, one way of guaranteeing this condition is by never increasing the step size parameter at successful iterations.

Second, there is the imposition of the so-called complete polling at all successful iterations. Complete polling requires the new iterate generated in the poll step to minimize the function in the poll set:

$$f(x_{k+1}) = f(x_k + \alpha_k d_k) \leq f(x_k + \alpha_k d) \quad \forall d \in D_k.$$

Complete polling necessarily costs $|D_k|$ (with $|D_k| \geq n+1$) function evaluations at every poll step, and not only at unsuccessful poll steps like in regular polling. The new iterate x_{k+1} could also be computed in a search step as long as $f(x_{k+1}) \leq f(x_k + \alpha_k d)$, for all $d \in D_k$, which means that the search step would have then to follow the (complete) poll step.

The proof of (7.10) under these two modifications is omitted. It can be accomplished in two phases. In a first phase, it is proved that for any $\epsilon > 0$ there exist $\alpha^-, \eta > 0$ such that $f(x_k + \alpha_k d_k) \leq f(x_k) - \eta \alpha_k \|\nabla f(x_k)\|$ if $\|\nabla f(x_k)\| > \epsilon$ and $\alpha_k < \alpha^-$. This inequality can be interpreted as a sufficient-decrease-type condition (see Chapters 9 and 10). A second phase consists of applying the Thomas argument [214] known for deriving lim-type results for trust-region methods (i.e., convergence results for the whole sequence of iterates; see also Chapter 10). The details are in [145].

7.6 The mesh adaptive direct-search method

Audet and Dennis introduced in [19] a class of direct-search algorithms capable of achieving global convergence in the nonsmooth case. This class of methods is called mesh adaptive direct search (MADS) and can be seen as an instance of Algorithm 7.2.

The globalization is achieved by simple decrease with integer lattices. So, let the mesh M_k (given, for instance, as in (7.8) but always by means of a finite D) be defined by Assumptions 7.8 and 7.9. Also let α_k be updated following Assumption 7.10. The key point in MADS is that $\mathcal{D}$ is allowed to be infinite, and thus different from the finite set D—which is important to allow some form of stationarity in the limit for the nonsmooth case—while the poll set P_k (defined in (7.3)) is still defined as a subset of the mesh M_k. (An earlier approach also developed to capture a rich set of directions can be found in [10].)

Thus, MADS first performs a search step by evaluating the objective function at a finite number of points in the mesh M_k. If the search step fails or is skipped, a poll set is tried by evaluating the objective function at the poll set P_k defined by the positive basis D_k chosen from a set of positive bases $\mathcal{D}$ (which is not necessarily explicitly given). However, this set $\mathcal{D}$ is now defined so that the elements $d_k \in D_k$ satisfy the following conditions:

- d_k is a nonnegative integer combination of the columns of D.

- The distance between x_k and the point $x_k + \alpha_k d_k$ tends to zero if and only if α_k does:

$$\lim_{k \in K} \alpha_k \|d_k\| \; = \; 0 \iff \lim_{k \in K} \alpha_k \; = \; 0 \tag{7.11}$$

  for any infinite subsequence K.

- The limits of all convergent subsequences of $\bar{D}_k = \{d_k / \|d_k\| : \; d_k \in D_k\}$ are positive bases.

In the spirit of the presentation in [19] we now define the concepts of refining subsequence and refining direction.

Definition 7.9. *A subsequence $\{x_k\}_{k \in K}$ of iterates corresponding to unsuccessful poll steps is said to be a refining subsequence if $\{\alpha_k\}_{k \in K}$ converges to zero.*

Let x be the limit point of a convergent refining subsequence. If the limit $\lim_{k \in L} d_k / \|d_k\|$ exists, where $L \subseteq K$ and $d_k \in D_k$, then this limit is said to be a refining direction for x.

The existence of a convergent refining subsequence is nothing else than a restatement of Corollary 7.2. It is left as an exercise to confirm that this result is still true for MADS. The next theorem states that the Clarke generalized derivative is nonnegative along any refining direction for x_* (the limit point of Corollary 7.2).

Theorem 7.10. *Let Assumptions* 7.1, 7.7, 7.8, 7.9, *and* 7.10 *hold. Then the sequence of iterates $\{x_k\}$ generated by MADS has a limit point x_* (given in Corollary* 7.2*) for which*

$$f^\circ(x_*; v) \; \geq \; 0$$

for all refining directions v for x_*.

Proof. Let $\{x_k\}_{k \in K}$ be the refining subsequence converging to x_* guaranteed by Corollary 7.2, and let $v = \lim_{k \in L} d_k / \|d_k\|$ be a refining direction for x_*, with $d_k \in D_k$ for all $k \in L$. Since f is Lipschitz continuous near x_* and $d_k / \|d_k\| \to v$ and $\alpha_k \|d_k\| \to 0$, for all $k \in L$,

$$f^\circ(x_*; v) \; \geq \; \limsup_{k \in L} \frac{f(x_k + \alpha_k \|d_k\| \frac{d_k}{\|d_k\|}) - f(x_k)}{\alpha_k \|d_k\|}. \tag{7.12}$$

Since x_k is an unsuccessful poll step,

$$\limsup_{k \in L} \frac{f(x_k + \alpha_k d_k) - f(x_k)}{\alpha_k \|d_k\|} \; \geq \; 0,$$

and the proof is completed. $\square$

Audet and Dennis [19] (see also [15]) proposed a scheme to compute the positive bases D_k, called lower triangular matrix based mesh adaptive direct search (LTMADS), which produces a set of refining directions for x_* with union asymptotically dense in $\mathbb{R}^n$ with probability one. From this and Theorem 7.10, MADS is thus able to converge to a point where the Clarke generalized directional derivative is nonnegative for a set of directions dense a.e. in $\mathbb{R}^n$, and not just for a finite set of directions as in Theorem 7.5. And more recently, Abramson et al. [8] proposed an alternative scheme to generate the positive bases D_k (also related to [10]), called OrthoMADS. This strategy also generates an asymptotically dense set of directions, but in a deterministic way, and each positive basis D_k is constructed from an orthogonal basis, thus determining relatively efficiently a reduction of the unexplored regions.

7.7 Imposing sufficient decrease

One alternative to the integrality requirements of Assumption 7.8, which would still provide global convergence for directional direct search, is to accept new iterates only if they satisfy a sufficient decrease condition. We will assume—in this section—that a new point $x_{k+1} \neq x_k$ is accepted (both in search and poll steps) only if

$$f(x_{k+1}) \ < \ f(x_k) - \rho(\alpha_k), \tag{7.13}$$

where the *forcing function* $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ is continuous, positive, and satisfies

$$\lim_{t \longrightarrow 0^+} \frac{\rho(t)}{t} = 0 \qquad \text{and} \qquad \rho(t_1) \ \leq \ \rho(t_2) \quad \text{if} \quad t_1 \ < \ t_2.$$

A simple example of a forcing function is $\rho(t) = t^2$. Functions of the form $\rho(t) = t^{1+a}$, for $a > 0$, are also in this category.

Theorem 7.11. *Suppose Algorithm* 7.2 *is modified in order to accept new iterates only if* (7.13) *holds.*

Let Assumption 7.1 *hold. If there exists an* $\alpha > 0$ *such that* $\alpha_k > \alpha$*, for all* k*, then the algorithm visits only a finite number of points. (In other words, Assumption* 7.2 *is satisfied.)*

Proof. Since ρ is monotonically increasing, we know that $0 < \rho(\alpha) \leq \rho(\alpha_k)$ for all $k \in \mathbb{N}_0$.

Suppose that there exists an infinite subsequence of successful iterates. From inequality (7.13) we get, for all successful iterations, that

$$f(x_{k+1}) \ < \ f(x_k) - \rho(\alpha_k) \ \leq \ f(x_k) - \rho(\alpha).$$

Recall that at unsuccessful iterations $f(x_{k+1}) = f(x_k)$. As a result, the sequence $\{f(x_k)\}$ must converge to $-\infty$, which clearly contradicts Assumption 7.1. ☐

To prove a result of the type of Theorem 7.3 for Algorithm 7.2, under the modification given in (7.13), we need to show first that for unsuccessful poll steps k_i one has

$$\|\nabla f(x_{k_i})\| \ \leq \ \left(\frac{\nu}{2} \mathrm{cm}(D_{k_i})^{-1} \max_{d \in D_{k_i}} \|d\| \right) \alpha_{k_i} + \frac{\mathrm{cm}(D_{k_i})^{-1}}{\min_{d \in D_{k_i}} \|d\|} \frac{\rho(\alpha_{k_i})}{\alpha_{k_i}}, \tag{7.14}$$

which is left as an exercise. Now, given the properties of the forcing function ρ, it is clear that $\nabla f(x_{k_i}) \to 0$ when $\alpha_{k_i} \to 0$, provided the minimum size of the vectors in D_k does not approach zero.

When imposing sufficient decrease, one can actually prove that the whole sequence of step size parameters converges to zero.

Theorem 7.12. *Suppose Algorithm* 7.2 *is modified in order to accept new iterates only if* (7.13) *holds.*

Let Assumption 7.1 *hold. Then the sequence of step size parameters satisfies*

$$\lim_{k \longrightarrow +\infty} \alpha_k = 0.$$

The proof is also left as an exercise. Note that Assumption 7.1 is more than what is necessary to prove Theorems 7.11 and 7.12. In fact, it would had been sufficient to assume that f is bounded from below on $L(x_0)$.

7.8 Other notes and references

The first reference in the literature to direct search has been attributed to a 1952 report of Fermi and Metropolis [90], in a form that resembles coordinate search (see the preface in Davidon [73]). In the 1950s, Box [42] and Box and Wilson [44] introduced direct-search algorithms related to coordinate search, using positive integer combinations of $D_\oplus$. Their algorithms evaluated points in M_k but not necessarily in P_k. Some of the vocabulary used at this time (like *two-level factorial and composite designs* [44]) was inspired from statistics where much of the early work on direct search was developed.[12]

Hooke and Jeeves [130] seemed to have been the first to use the terminology *direct-search methods*. Hooke and Jeeves [130] are also acknowledged to have been the first to recognize the underlying notion of pattern or integer lattice in direct search, which was then explored by other authors, in particular by Berman [35]. Hooke and Jeeves' *exploratory moves* scheme is a predecessor of the search step. Later, in the 1990s, Torczon [216, 217] showed how to rigorously use integer lattices in the globalization of *pattern-search methods* (which can be defined as directional direct-search methods based on such lattices, as explained in Section 7.5). Audet and Dennis [18] contributed to the field by essentially focusing the analysis of these methods on the subsequence of unsuccessful iterates. The paper of Booker et al. [40] should get most of the credit for the formal statement of the search-poll framework.

But the pioneer work on direct search was not confined to directional methods based on patterns generated by fixed sets of directions. In fact, some of the earliest directional direct-search methods modified the search directions at the end of each iteration by combining, in some form, a previously computed set of directions. Among such methods are the ones by Powell [183] which used conjugate directions (see also the modifications introduced by Zangwill [237] and the analysis in Toint and Callier [215]) and by Rosenbrock [201]. A recent approach has been pursued by Frimannslund and Steihaug [100] by explicitly rotating the direction set based on curvature information extracted from function values.

The introduction of a sufficient decrease condition (involving the step size) in direct search was first made by Yu [235] in 1979. Other authors have explored the use of such a condition in directional direct-search methods, like Grippo, Lampariello, and Lucidi [115], Lucidi and Sciandrone [160], and García-Palomares and Rodríguez [103]. The work of Lucidi and Sciandrone [160], in particular, concerns the development of an algorithmic framework, exploring the use of line-search techniques in directional direct-search

[12] J. A. Nelder and R. Mead were also statisticians.

methods. Their convergence theory includes first-order lim-type results derived under reasonable assumptions. A particularity of the approaches in [103, 160] is the consideration of different step sizes along different directions. Diniz-Ehrhardt, Martínez, and Raydan [79] used a sufficient decrease condition in the design of a nonmonotone algorithm.

In the context of globalization of directional direct-search methods by integer lattices (see Section 7.5), it is possible in suitable cases to relax the assumption that the directions are extracted from a finite set D. This has been explored by Coope and Price [66] in their grid-based methods. They have observed that after an unsuccessful iteration one can change D (provided it still satisfies Assumption 7.8) and, thus, gain further flexibility in attempting to capture the curvature of the function. However, there is a price to pay, namely, that $\alpha_k \to 0$ should be imposed, which, for example, can be guaranteed in the context of Theorems 7.1 and 7.8 by never allowing α_k to increase.

There has been some effort in trying to develop efficient serial implementations of pattern-search methods by considering particular instances where the problem structure can be exploited efficiently. Price and Toint [195] examined how to take advantage of partial separability. Alberto et al. [10] have shown ways of incorporating user-provided function evaluations. Abramson, Audet, and Dennis [6] looked at the case where some incomplete form of gradient information is available. Custódio and Vicente [70] suggested several procedures, for general objective functions, to improve the efficiency of pattern-search methods using simplex derivatives. In particular, they showed that ordering the poll directions in opportunistic polling according to a negative simplex gradient can lead to a significant reduction in the overall number of function evaluations (see [68, 70]).

One attractive feature of directional direct-search methods is that it is easy to parallelize the process of evaluating the function during poll steps. Many authors have experimented with different parallel versions of these methods; see [10, 21, 77, 103, 132]. Asynchronous parallel forms of these methods have been proposed and analyzed by García-Palomares and Rodríguez [103] and Hough, Kolda, and Torczon [132] (see also the software produced by Gray and Kolda [110]).

Another attractive feature is the exploration of the directionality aspect to design algorithms for nonsmooth functions with desirable properties. We have mentioned in Section 7.6 that the MADS methods can converge with probability one to a first-order stationary, nonsmooth point. It is shown in [3] how to generalize this result to second-order stationary points with continuous first-order derivatives but nonsmooth second-order derivatives. Other direct-search approaches to deal with nonsmooth functions have recently been proposed [24, 37] but for specific types of nondifferentiability.

The generating search set (GSS) framework

Kolda, Lewis, and Torczon [145] introduced another framework for globally convergent directional direct-search methods. These authors do not make an explicit separation in their algorithmic description between the search step and the poll step. A successful iterate in their GSS framework is of the form $x_k + \alpha_k d_k$, where d_k belongs to a set of directions $G_k \cup H_k$. In GSS, G_k plays the role of our D_k (used in the poll step). The search step is accommodated by the additional set of directions H_k (which might, as in the framework presented in Section 7.2, be empty). When the iterates are accepted solely based on simple decrease of the objective function, integrality requirements similar to those of

Assumption 7.8 (or something equivalent) must be imposed on the finite set of directions that contains all choices of $G_k \cup H_k$ for all $k \in \mathbb{Z}_+$.

The multidirectional search method

Another avenue followed in the development of direct search is simplicial methods (like the Nelder and Mead simplex method [177]). Simplicial direct-search methods, despite sharing features with directional direct search, have their own motivation and, thus, will be treated separately in Chapter 8.

The multidirectional search (MDS) method of Dennis and Torczon [77], described next, can be regarded as both a directional and a simplicial direct-search method. We choose to include MDS in the book essentially for historical reasons and because it will help us frame the modifications necessary to make the Nelder–Mead method globally convergent.

As in the Nelder–Mead method (whose details are not needed now), MDS starts with a simplex of $n+1$ vertices $Y = \{y^0, y^1, \ldots, y^n\}$. Each iteration is centered at the simplex vertex y^0 with the lowest function value (in contrast with Nelder–Mead which focuses particular attention at the vertex with the highest function value). Then a rotated simplex is formed by rotating the vertices y^i, $i = 1, \ldots, n$, 180° around y^0 (see Figure 7.7). (The reader might have already identified a maximal positive basis...) If the best objective value of the rotated vertices is lower than $f(y^0)$, then an expanded simplex is formed in the direction of the rotated one (see Figure 7.7). The next iteration is started from either the rotated or expanded simplex, depending on which is better. If the best objective value of the rotated vertices is no better than $f(y^0)$, then a shrink step is taken just as in Nelder–Mead (see Figure 7.7), and the next iteration is started from the shrunken simplex. We now give more details on the MDS algorithm.

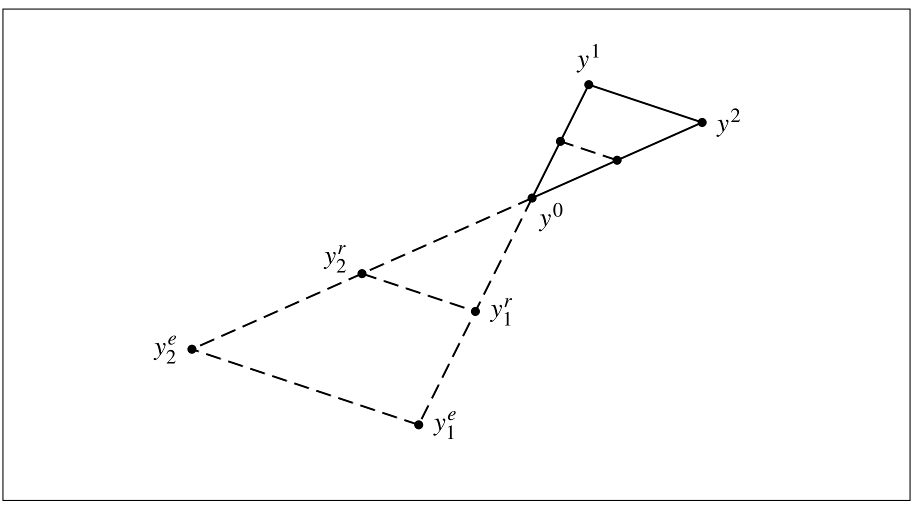

Figure 7.7. *Original simplex, rotated vertices, expanded vertices, shrunken vertices, corresponding to an MDS iteration.*

Algorithm 7.3 (The MDS method).

Initialization: Choose an initial simplex of vertices $Y_0 = \{y_0^0, y_0^1, \ldots, y_0^n\}$. Evaluate f at the points in Y_0. Choose constants:

$$0 < \gamma^s < 1 < \gamma^e.$$

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

0. Set $Y = Y_k$.
1. **Find best vertex:** Order the $n+1$ vertices of $Y = \{y^0, y^1, \ldots, y^n\}$ so that $f^0 = f(y^0) \leq f(y^i)$, $i = 1, \ldots, n$.
2. **Rotate:** Rotate the simplex around the best vertex y^0:

$$y_i^r = y^0 - (y^i - y^0), \quad i = 1, \ldots, n.$$

   Evaluate $f(y_i^r)$, $i = 1, \ldots, n$, and set $f^r = \min\{f(y_i^r) : i = 1, \ldots, n\}$. If $f^r < f^0$, then attempt an expansion (and then take the best of the rotated or expanded simplices). Otherwise, contract the simplex.

3. **Expand:** Expand the rotated simplex:

$$y_i^e = y^0 - \gamma^e (y^i - y^0), \quad i = 1, \ldots, n.$$

   Evaluate $f(y_i^e)$, $i = 1, \ldots, n$, and set $f^e = \min\{f(y_i^e) : i = 1, \ldots, n\}$. If $f^e < f^r$, then accept the expanded simplex and terminate the iteration: $Y_{k+1} = \{y^0, y_1^e, \ldots, y_n^e\}$. Otherwise, accept the rotated simplex and terminate the iteration: $Y_{k+1} = \{y^0, y_1^r, \ldots, y_n^r\}$.

4. **Shrink:** Evaluate f at the n points $y^0 + \gamma^s (y^i - y^0)$, $i = 1, \ldots, n$, and replace $y^1, \ldots, y^n$ by these points, terminating the iteration: $Y_{k+1} = \{y^0 + \gamma^s (y^i - y^0),\ i = 0, \ldots, n\}$.

Typical values for γ^s and γ^e are $1/2$ and 2, respectively. A stopping criterion could consist of terminating the run when the diameter of the simplex becomes smaller than a chosen tolerance $\Delta_{tol} > 0$ (for instance, $\Delta_{tol} = 10^{-5}$).

Torczon [216] noted that, provided γ^s and γ^e are rational numbers, all possible vertices visited by the algorithm lie in an integer lattice. This property is independent of the position in each simplex taken by its best vertex. In addition, note that the MDS algorithm enforces a simple decrease to accept new iterates (otherwise, the simplex is shrunk and the best vertex is kept the same). Thus, once having proved the integer lattice statement, the proof of the following theorem follows trivially from the material of Section 7.3.

Theorem 7.13. *Suppose that $\gamma^s, \gamma^e \in \mathbb{Q}$, and let the initial simplex be of the form $Y_0 = G\bar{Z}$, where $G \in \mathbb{R}^{n \times n}$ is nonsingular and the components of $\bar{Z} \in \mathbb{R}^{n \times (n+1)}$ are integers. Assume that $L(y_0^0) = \{x \in \mathbb{R}^n : f(x) \leq f(y_0^0)\}$ is compact and that f is continuously differentiable in $L(y_0^0)$. Then the sequence of iterates $\{y_k^0\}$ generated by the MDS method (Algorithm 7.3) has one stationary limit point x_*.*

Proof. We need to convince the reader that we can frame MDS in the format of directional direct search (Algorithm 7.2). Notice, first, that the expanded step can be seen as a search step. Polling is complete and involves a maximal positive basis D_k related to the initial simplex and chosen from the set D formed by

$$\left\{ y_0^j - y_0^i,\ j = 0,\ldots,n,\ j \neq i \right\} \cup \left\{ -(y_0^j - y_0^i),\ j = 0,\ldots,n,\ j \neq i \right\},$$

$i = 0,\ldots,n$. It is then a simple matter to see that the integer lattice requirements (see Section 7.5), i.e., Assumptions 7.8, 7.9, and 7.10, are satisfied. $\square$

Given the pointed nature of the meshes generated by Algorithm 7.3, it is not necessary that Y_0 takes the form given in Theorem 7.13. In fact, the original proof in [216] does not impose this assumption. We could also have lifted it here, but that would require a modification of the mesh/grid framework of Section 7.5.

Another avenue to make MDS globally convergent to stationary points is by imposing sufficient decrease in the acceptance conditions, as is done in Chapter 8 for the modified Nelder–Mead method.

7.9 Exercises

1. In the context of the globalization of the directional direct-search method (Algorithm 7.2) with simple decrease with integer lattices (Section 7.5), prove that the mesh M_k defined by (7.8) can be generalized to

$$M_k = \bigcup_{x \in \mathcal{E}_k} \{x + \alpha_k D u : u \in \mathbb{Z}_+^{|D|}\}, \tag{7.15}$$

   where $\mathcal{E}_k$ is the set of points where the objective function f has been evaluated by the start of iteration k (and $\mathbb{Z}_+$ is the set of nonnegative integers).

2. Let the mesh be defined by Assumptions 7.8 and 7.9 for the MADS methods. Let α_k be updated following Assumption 7.10. Prove under Assumption 7.1 that the result of Corollary 7.2 is true (in other words that there exists a convergent refining subsequence).

3. Show (7.12). You will need to add and subtract a term and use the Lipschitz continuity of f near x_*.

4. Generalize Theorem 2.8 for unsuccessful poll steps when a sufficient decrease condition of the form (7.13) is imposed. Show that what you get is precisely the bound (7.14).

5. Prove Theorem 7.12.