ON THE LOCAL CONVERGENCE OF PATTERN SEARCH

ELIZABETH D. DOLAN*, ROBERT MICHAEL LEWIS†, AND VIRGINIA TORCZON‡

Abstract. We examine the local convergence properties of pattern search methods, complementing the previously established global convergence properties for this class of algorithms. We show that the step-length control parameter which appears in the definition of pattern search algorithms provides a reliable asymptotic measure of first-order stationarity. This gives an analytical justification for a traditional stopping criterion for pattern search methods. Using this measure of first-order stationarity, we both revisit the global convergence properties of pattern search and analyze the behavior of pattern search in the neighborhood of an isolated local minimizer.

Key words. pattern search, local convergence analysis, global convergence analysis, stopping criteria, desultory rate of convergence

1. Introduction.

Pattern search methods are a class of direct search methods for solving nonlinear optimization problems. In a series of papers [16, 11, 12, 13, 14] we established the global convergence properties of pattern search for both constrained and unconstrained problems. In this paper, we consider the local convergence properties of pattern search and revisit the global convergence properties in light of these new results.

For simplicity, our discussion will focus on the case of unconstrained minimization:

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

where $f : \mathbb{R}^n \to \mathbb{R}$. Results similar to those we present here also can be derived for the general case of bound and linear constraints [12, 13]. However, the underlying ideas are simpler to explain for the unconstrained case.

We first show how the pattern size parameter, which plays a central role in the definition of pattern search methods and tacitly serves as a step-length control mechanism, also provides a reliable asymptotic measure of first-order stationarity. This gives an analytical justification for the traditional use of the pattern size parameter as a stopping criterion. We also establish a local convergence result concerning the behavior of the sequence of iterates produced by a pattern search algorithm in the

*Industrial Engineering and Management Sciences, Northwestern University and Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois 60439–4844; dolan@mcs.anl.gov. This research was supported by the National Science Foundation under Grant CCR–9734044 while the author was in residence at the College of William & Mary; by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-Eng-38; and by the National Science Foundation (Challenges in Computational Science Grant CDA-9726385 and (Information Technology Research) Grant CCR-0082807.

†Department of Mathematics, College of William & Mary, P. O. Box 8795, Williamsburg, Virginia 23187–8795; buckaroo@math.wm.edu. This research was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1–97046 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, Virginia 23681–2199 and by the Computer Science Research Institute at Sandia National Laboratories.

‡Department of Computer Science, College of William & Mary, P. O. Box 8795, Williamsburg, Virginia 23187–8795; va@cs.wm.edu. This research was supported by the National Science Foundation under Grant CCR–9734044, by the National Aeronautics and Space Administration under NASA Contract No. NAS1–97046 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, Virginia 23681–2199, and by the Computer Science Research Institute at Sandia National Laboratories.

neighborhood of an isolated local minimizer x_*. These analytical results are illustrated with some simple numerical experiments on quadratic objectives.

What is interesting about the analysis presented here is that we can establish local convergence properties despite the fact that direct search methods do not employ an explicit representation of the gradient of the objective and, as a consequence, cannot enforce a notion of sufficient decrease. We proved global convergence results for pattern search by showing that all iterates lie on a rational lattice. It is this restriction on the form of the steps that allows us to relax the notion of sufficient decrease and yet still prove global convergence. Pattern search may accept any point on the current integer lattice so long as it produces simple decrease on the value of the objective function at the current iterate. However, key to the global analysis is the notion of having searched in a sufficient number of directions from the current iterate to guarantee that we have not overlooked a potential direction of descent. It is only after searching over a sufficient set of directions that we are allowed to reduce the current step-length control parameter—which has the effect of refining the lattice over which we are searching.

This notion of sufficient local information at iterations at which we reduce the step-length control parameter allows us to show that the pattern size, as measured by the step-length control parameter, provides a reliable asymptotic measure of first-order stationarity. This analytical result is gratifying since it vindicates the long-standing use of the step-length control parameter as a stopping criterion for direct search methods (see, for instance, Section 4 of [8]). The result on the correlation of the step-length control parameter and stationarity then enables us to study the local convergence properties of pattern search.

Notation. We use $\mathcal{L}(x_0)$ to denote the set $\{ x \mid f(x) \leq f(x_0) \}$. We use ∂ to denote the boundary of a given set. It is assumed, unless otherwise noted, that all norms are Euclidean vector norms or the associated operator norm. Given x and $r > 0$, we denote by $\mathcal{B}(x, r)$ the open ball of radius r centered at x so that $\mathcal{B}(x, r) = \{ y \mid \| y - x \| < r \}$. We also acknowledge an abuse of notation that is nonetheless convenient: if y is a vector and A is a matrix, we use the notation $y \in A$ to mean that the vector y is contained in the set of columns of the matrix A.

2. Pattern search. We first review the elements of pattern search that play a role in our local analysis. There are rigorous formal definitions of pattern search [16, 11], several features of which we will recall shortly. However, pattern search can perhaps be most quickly understood with the following simple example of a pattern search algorithm. At iteration k, we have an iterate $x_k \in \mathbb{R}^n$ and a step-length control parameter $\Delta_k > 0$. Let e_j, $j = 1, \ldots, n$ be the standard unit basis vectors. For the purposes of this example, we represent the set of directions that we will use for the search as the set $\mathcal{D} \equiv \{d_i\}_{i=1}^{2n} \equiv \{e_1, \ldots, e_n, -e_1, \ldots, -e_n\}$ though, as we discuss shortly, many other choices are possible. We now have several algorithmic options open to us. We consider the simple opportunistic strategy, which is to look successively at the points $x_+ = x_k + \Delta_k d_i$, $i \in \{1, \ldots, 2n\}$ until either we find an x_+ for which $f(x_+) < f(x_k)$ or we exhaust all $2n$ possibilities. Figure 2.1 illustrates the pattern of points among which we search for x_+ when $n = 2$.

If we find no x_+ such that $f(x_+) < f(x_k)$, then we call the iteration *unsuccessful*; otherwise, we consider the iteration *successful* since we have found a new iterate that produces decrease on f at x_k. When the iteration is unsuccessful, we set $x_{k+1} = x_k$ and are required to reduce Δ_k (typically, by a half) before continuing; otherwise, for a successful iteration, we set $x_{k+1} = x_+$ and leave the step-length control parameter

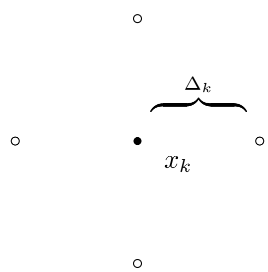

FIG. 2.1. *A simple instance of a pattern in* $\mathbb{R}^2$.

alone, i.e., $\Delta_{k+1} = \Delta_k$ (though the analysis also allows us to increase Δ_k if the iteration is a success). We repeat this process until some suitable stopping criterion is satisfied.

Note that overall our requirements on the outcome of the search at each iteration are light: if after searching over all the points defined by $\Delta_k d_i$, $i = 1, \ldots, 2n$ we fail to find a point $x_+ = x_k + \Delta_k d_i$ that reduces the value of f at x_k, then we must try again with a smaller value of Δ_k. Otherwise, we accept as our new iterate the first point in the pattern that produces decrease. In the latter case, we may choose to increase Δ_k. In either case, we are free to make changes to the set of search directions $\mathcal{D}$ to be used in the next iteration, though we leave $\mathcal{D}$ unchanged in the example given previously. In general, changes to either the step-length control parameter or the set of search directions are subject to certain algebraic conditions, outlined fully in [11].

A distinguishing characteristic of pattern search methods is that they sample the function over a predefined pattern of points, all of which lie on a rational lattice. By enforcing structure on the form of the points in the pattern, as well as some simple rules on both the outcome of the search and the subsequent updates, standard global convergence results can be obtained [16, 11].

There remains the question of what constitutes an acceptable set of search directions. A pattern must form a positive spanning set for $\mathbb{R}^n$ [5]. A set of vectors $\{a_1, \ldots, a_p\}$ *positively spans* $\mathbb{R}^n$ if any vector $x \in \mathbb{R}^n$ can be written as a nonnegative linear combination of the vectors in the set; i.e.,

$$x \quad = \quad \alpha_1 a_1 + \cdots + \alpha_p a_p \quad \alpha_i \geq 0 \quad \forall i.$$

The set $\{a_1, \ldots, a_p\}$ is called *positively dependent* if one of the a_i's is a nonnegative combination of the others; otherwise the set is *positively independent*. A *positive basis* is a positively independent set whose positive span is $\mathbb{R}^n$.

It is straightforward to verify that the set of vectors $\{e_1, \ldots, e_n, -e_1, \ldots, -e_n\}$ we used to define the pattern for our simple example is a positive spanning set.

2.1. Prior results. Before proceeding to our local convergence results, we recall the following proposition from [11], which we state here without proof.

PROPOSITION 2.1. *Given any set* $\{a_1, \ldots, a_r\}$ *that positively spans* $\mathbb{R}^n$, $a_i \neq 0$ *for* $i = 1, \ldots, r$, *there exists* $c_{2.1} > 0$ *such that for all* $x \in \mathbb{R}^n$, *we can find an* a_i *for which*

$$x^T a_i \geq c_{2.1} \| x \| \, \| a_i \|.$$

Note that this is a purely geometric property of positive spanning sets.

2.2. Some formal definitions. We also need to recall some notation regarding both the pattern and the form of the search. For the details, we refer the reader to [16, 11].

We have noted already that the pattern must form a positive spanning set for $\mathbb{R}^n$. In fact, we represent the pattern using two components, a *basis matrix* and a *generating matrix*.

The basis matrix can be any nonsingular matrix $B \in \mathbb{R}^{n\times n}$.

The generating matrix is an integral matrix $C_k \in \mathbb{Z}^{n\times p_k}$, where $p_k > n+1$. We require C_k to contain a minimum of $n+2$ columns because the minimum number of vectors in a positive spanning set is $n+1$ [5]; for convenience, we require a column of zeros to denote the zero step. We further partition the generating matrix to reveal the positive basis that guarantees that the pattern positively spans $\mathbb{R}^n$. We call the columns associated with the positive basis the *core pattern*, which we denote Γ_k; any remaining columns in the positive spanning set are denoted L_k:

$$C_k = \begin{bmatrix} \Gamma_k & L_k & 0 \end{bmatrix}. \tag{2.1}$$

We further require that $\Gamma_k \in \mathbf{\Gamma}$, where $\mathbf{\Gamma}$ comprises a finite set of integral matrices, the columns of which form a positive basis for $\mathbb{R}^n$.

A *pattern* is then represented by the columns of the matrix $P_k = BC_k$. For convenience, we use the partition of the generating matrix C_k given in (2.1) to partition P_k as follows:

$$P_k = BC_k = \begin{bmatrix} B\Gamma_k & BL_k & 0 \end{bmatrix}.$$

To tie this notation back to the example that introduces Section 2, we note that $B = I$, $\Gamma_k = [\, I \ -I \,]$ and $L_k \equiv \emptyset$. Since the choices of Γ_k and L_k are fixed in our example, $P_k \equiv [\, I \ -I \ \ 0 \,]$ for all k.

Now, given the step-length control parameter $\Delta_k \in \mathbb{R}$, $\Delta_k > 0$, we define a *trial step* s_k^i to be any vector of the form $s_k^i = \Delta_k B c_k^i$, where c_k^i is a column of C_k.

In Figure 2.2 we state the general form of a pattern search method for unconstrained minimization.

Let $x_0 \in \mathbb{R}^n$ and $\Delta_0 > 0$ be given.
For $k = 0, 1, \ldots$, until convergence do:

1. Compute $f(x_k)$.
2. Determine a step s_k using an unconstrained exploratory moves algorithm.
3. If $f(x_k + s_k) < f(x_k)$, then $x_{k+1} = x_k + s_k$. Otherwise $x_{k+1} = x_k$.
4. Update C_k and Δ_k.

FIG. 2.2. *Generalized pattern search for unconstrained minimization.*

We have remarkable latitude in our choice of the step s_k. For the global convergence analysis to hold, we need only satisfy the hypotheses on the outcome of the unconstrained exploratory moves, given in Figure 2.3.

1. $s_k \in \Delta_k P_k$.
2. If $\min \{ \, f(x_k + y) \ | \ y \in \Delta_k B\Gamma_k \, \} < f(x_k)$, then $f(x_k + s_k) < f(x_k)$.

FIG. 2.3. *Hypotheses on the outcome of the unconstrained exploratory moves.*

A few comments on these hypotheses are in order. The first hypothesis is straightforward: the step returned must be a column in the current pattern matrix P_k, scaled by the current value of the step-length control parameter Δ_k. This condition ensures that the steps we consider remain on the rational lattice; arbitrary steps are not allowed.

For our purposes, the second hypothesis is the more interesting. Notice that in Figure 2.2, a successful iteration of pattern search requires only that the step s_k produce simple decrease, i.e., $f(x_k + s_k) < f(x_k)$. Thus, *any* nonzero step defined by a column of $\Delta_k P_k$ that satisfies the condition $f(x_k + s_k) < f(x_k)$ may be returned by the exploratory moves since it immediately satisfies both of the hypotheses given in Figure 2.3—even if we do not explicitly find $\min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k \}$.

The second hypothesis in Figure 2.3 ensures that we have sufficient information about the local behavior of f to declare an iteration *unsuccessful*, accept the zero step $s_k = 0$ (so that $x_{k+1} \equiv x_k$), and reduce Δ_k to continue the search with smaller steps at the next iteration. The second hypothesis implicitly decrees that we may only return the zero step, and thus reduce Δ_k, when we have looked at all the steps defined by the core pattern, i.e., all steps of the form $y \in \Delta_k B \Gamma_k$.

The core pattern $B\Gamma_k$ must be a positive basis. This means that even though we do not have an explicit representation of $\nabla f(x_k)$ (assuming that f is differentiable), the geometric property of positive spanning sets captured in Proposition 2.1 gives us a positive lower bound, which is independent of k, on the angle between $-\nabla f(x_k)$ (assuming it is nonzero) and some a_i in the positive spanning set. At any given iteration, we do not know for which a_i this lower bound holds. However, this guaranteed lower bound, when combined with the second hypothesis in Figure 2.3, ensures that at the end of an unsuccessful iteration, we have significant information about the local behavior of f at x_k. Furthermore, the quality of our local information improves as we reduce Δ_k.

Finally, we make a brief comment on the basic rules for updating Δ_k, which are given in Figure 2.4. We also must impose additional conditions on the choice of θ and

1. If all $f(x_k + s_k) \geq f(x_k)$, then $\Delta_{k+1} = \theta \Delta_k$, where $\theta \in (0, 1)$.
2. If any $f(x_k + s_k) < f(x_k)$, then $\Delta_{k+1} = \lambda_k \Delta_k$, where $\lambda_k \geq 1$.

FIG. 2.4. *Basic rules for updating* Δ_k .

λ_k to ensure that Theorem 3.2 from [16] holds. Rather than detail these conditions here, since they are outlined fully in [16] (other options are discussed in [10]), we note the two essential consequences. First, if our choices for θ and λ_k ensure that Theorem 3.2 from [16] holds, then all the iterates lie on a translated integer lattice. Second, the rules for updating Δ_k ensure that Δ_k is *reduced* after any unsuccessful iteration since $\theta \in (0, 1)$. The latter means that after any unsuccessful iteration, pattern search refines the lattice of points over which the search resumes.

We can capitalize on the structure of pattern search refinement to construct local convergence results. The subsequence of *unsuccessful* iterates, which is what interests us here, is well-defined: they are the iterates at which we must reduce Δ_k to ensure that the search can make further progress. We reduce Δ_k only after we have sufficient local information about the behavior of f to justify this action: we have considered all the steps defined by the columns of $\Delta_k \Gamma_k$ and none of them have produced descent on f at x_k. We presently use this fact to assess stationarity.

3. Measuring first-order stationarity. The following theorem shows that the step-length control parameter Δ_k, when small enough, provides a reasonable measure of first-order stationarity at an unsuccessful iterate. For simplicity, we assume that $\nabla f(x)$ is Lipschitz continuous. For the reader interested in greater generality, we note that a similar result can be proven under the assumption of uniform continuity.

THEOREM 3.1. *Suppose that for some $\rho > 0$, $\nabla f(x)$ is Lipschitz continuous, with Lipschitz constant $\mathcal{K}$, on the open neighborhood $\Omega = \cup_{x \in \mathcal{L}(x_0)} \mathcal{B}(x, \rho)$ of $\mathcal{L}(x_0)$. Then there exist $\delta_{3.1} > 0$ and $c_{3.1} > 0$ for which the following holds. If x_k is an unsuccessful iterate and $\Delta_k < \delta_{3.1}$, then*

$$\| \nabla f(x_k) \| \leq c_{3.1} \Delta_k.$$

Proof. Let $r = \frac{1}{2} \min\{1, \rho\}$. If $x \in \mathcal{L}(x_0)$, then the ball $\mathcal{B}(x, r)$ is contained in Ω. We are interested in steps of the form $s = \Delta_k B c_k^i$, where c_k^i is a column of the core matrix Γ_k. Since $\Gamma_k \in \mathbf{\Gamma}$ and $\mathbf{\Gamma}$ is finite, $\| s \| \leq \Delta_k \| B \| \Gamma^*$, where Γ^* is the maximum norm of any column of the matrices in the set $\mathbf{\Gamma}$. Set $\delta_{3.1} = r / (\| B \| \Gamma^*)$.

By the definition of pattern search, for any $\Gamma_k \in \mathbf{\Gamma}$ the set $\{ \, s \mid s \in \Delta_k B \Gamma_k \, \}$ forms a positive basis for $\mathbb{R}^n$. Thus Proposition 2.1 assures us of the existence of a step s for which

$$-\nabla f(x_k)^T s \geq c_{2.1} \| \nabla f(x_k) \| \, \| s \|. \tag{3.1}$$

Since iteration k is unsuccessful, it follows that

$$f(x_k + s) - f(x_k) \quad \geq \quad 0 \quad \forall \, s \in \Delta_k B \Gamma_k.$$

Since we assume $\Delta_k < \delta_{3.1}$, $(x_k + s) \in \mathcal{B}(x_k, r) \subset \Omega$, and we can apply the mean value theorem. In addition, using (3.1) and the Cauchy–Schwarz inequality, for some ξ in the line segment connecting x_k and $x_k + s$ we have

$$\begin{aligned} 0 &\leq f(x_k + s) - f(x_k) \\ &= \nabla f(x_k)^T s + (\nabla f(\xi) - \nabla f(x_k))^T s \\ &\leq -c_{2.1} \| \nabla f(x_k) \| \, \| s \| + \| \nabla f(\xi) - \nabla f(x_k) \| \, \| s \|, \end{aligned}$$

where s is the step for which (3.1) holds. Thus

$$c_{2.1} \| \nabla f(x_k) \| \leq \| \nabla f(\xi) - \nabla f(x_k) \|.$$

Again, since $\mathcal{B}(x_k, r) \subset \Omega$, the Lipschitz continuity of $\nabla f(x)$ gives us

$$c_{2.1} \| \nabla f(x_k) \| \quad \leq \quad \mathcal{K} \, \| \xi - x_k \| \quad \leq \quad \mathcal{K} \, \| s \| \quad \leq \quad \mathcal{K} \, \Delta_k \, \| B \| \, \Gamma^*.$$

Therefore

$$\| \nabla f(x_k) \| \leq c_{3.1} \Delta_k,$$

with $c_{3.1} = \mathcal{K} \, \| B \| \, \Gamma^* / c_{2.1}$. ☐

Theorem 3.1 gives a theoretical justification for a traditional stopping criterion for pattern search methods. In the long literature on direct search methods, one frequently encounters the suggestion that a direct search method be terminated when some measure of the step size first falls below a value deemed suitably small [8, 4, 2].

In the case of pattern search, Theorem 3.1 vindicates this intuition. At unsuccessful iterations, the step size in pattern search (as measured by Δ_k) provides a bound on first-order stationarity. At the same time, it is after the unsuccessful iterations that Δ_k is decreased. Thus, decrease in Δ_k provides a simple measure of progress which can be used reliably to test for convergence. We discuss further the use of Δ_k to measure progress when we present some numerical examples in Section 5.

A similar relation between Δ_k and constrained stationarity in the case of pattern search for bound constrained problems is explicitly used in the pattern search augmented Lagrangian algorithm in [14]. The result plays a critical role in allowing successive inexact minimization of an augmented Lagrangian without an explicit estimate of the gradient. A relation similar to Theorem 3.1 for linearly constrained pattern search appears in [13].

The global convergence analysis of pattern search in [16] says that if $\mathcal{L}(x_0)$ is compact, then $\liminf_{k\to\infty} \Delta_k = 0$ and $\liminf_{k\to\infty} \| \nabla f(x_k) \| = 0$. The former result and Theorem 3.1 allow us to sharpen the latter result. Let the set $\mathcal{U}$ represent a subsequence of unsuccessful iterates for which $\lim_{k\to\infty, k\in\mathcal{U}} \Delta_k = 0$ (such a subsequence exists since $\liminf_{k\to\infty} \Delta_k = 0$). Then Theorem 3.1 says that we have $\lim_{k\to\infty, k\in\mathcal{U}} \| \nabla f(x_k) \| = 0$.

The general result $\liminf_{k\to\infty} \| \nabla f(x_k) \| = 0$ for pattern search leaves open the possibility that $\| \nabla f(x_k) \|$ does not converge. In [1], Audet shows that this actually can occur by constructing a pattern search algorithm and an objective function for which $\{x_k\}$ has infinitely many limit points, one of which is not a stationary point of the objective. However, in his example, the subsequence of iterates converging to the non-stationary point of the objective function are all *successful* iterates. Theorem 3.1 reassures us that in practice we need not worry about convergence to non-stationary points. If we stop the algorithm at the first *unsuccessful* iterate for which $\Delta_k < \Delta_*$ for some suitably small stopping tolerance Δ_*, then Theorem 3.1 says that we may reasonably expect $\| \nabla f(x_k) \|$ to be small.

4. Local convergence. We now consider the local convergence of pattern search methods. We begin with a collection of hypotheses and definitions we will need.

The first condition is a mild hypothesis on the generating matrices C_k that allows us to bound the size of the steps $\{s_k\}$.

HYPOTHESIS 0. *The columns of the generating matrices $C_k = \left[c_k^1 \cdots c_k^{p_k}\right]$ remain bounded in norm, i.e., there exists $C_0 > 0$ such that for all k, $C_0 > \left\| c_k^i \right\|$, for all $i = 1, \cdots, p_k$. Thus, there exists a constant $c_0 > 0$ such that any step s_k satisfies*

$$\| s_k \| \leq c_0 \Delta_k.$$

We also impose the following condition on the step-length control parameter Δ_k.

HYPOTHESIS 1. *There exists N for which Δ_k is monotonically nonincreasing for all $k \geq N$.*

Note that this is a condition we can explicitly enforce by not allowing increases in Δ_k after some iteration N; Δ_k can stay the same or decrease.

The local convergence results are concerned with the behavior of pattern search in a neighborhood of an isolated local minimizer x_*. We make the following assumptions about the behavior of f in a neighborhood of x_*.

HYPOTHESIS 2. *We assume the existence of an open ball $\mathcal{B}(x_*, r)$, $r > 0$, for which $f(x)$ is twice continuously differentiable on $\mathcal{B}(x_*, r)$, $\nabla^2 f(x)$ is positive definite* *on $\mathcal{B}(x_*, r)$, $\nabla f(x_*) = 0$, and lower and upper bounds $\sigma_{\min}$ and $\sigma_{\max}$ on the singular values of $\nabla^2 f(x)$ on $\mathcal{B}(x_*, r)$ exist. We further assume $\sigma_{\min} > 0$.*

We then define

$$\kappa = \sigma_{\max}/\sigma_{\min}. \tag{4.1}$$

We also define

$$\eta = r/(\| B \| \Gamma^* + 1). \tag{4.2}$$

This choice ensures that if $\| x_k - x_* \| < \eta$ and $\Delta_k < \eta$, then for any step $s \in \Delta_k B \Gamma_k$ we have $\| (x_k + s) - x_* \| < r$.

Our first result relates Δ_k to $\| x_k - x_* \|$ at unsuccessful iterates.

PROPOSITION 4.1. *Under Hypothesis 2, there exists $c_{4.1} > 0$ for which the following holds. If x_k is an unsuccessful iterate, $\Delta_k < \eta$, and $\| x_k - x_* \| < \eta$ (where η is as in (4.2)), then*

$$\| x_k - x_* \| \leq c_{4.1} \Delta_k.$$

Proof. Proposition 2.1 assures us of the existence of a step $s \in \Delta_k B \Gamma_k$ for which

$$-\nabla f(x_k)^T s \geq c_{2.1} \| \nabla f(x_k) \| \, \| s \|. \tag{4.3}$$

If iteration k is unsuccessful, it follows that

$$f(x_k + s) - f(x_k) \quad \geq \quad 0 \quad \forall \ s \in \Delta_k B \Gamma_k.$$

Because $\Delta_k < \eta$, we know that $(x_k + s) \in \mathcal{B}(x_*, r)$, where f is differentiable and we can apply the mean value theorem. In addition, using (4.3) and the Cauchy–Schwarz inequality, for some ξ in the line segment connecting x_k and $x_k + s$ we have

$$0 \quad \leq \quad f(x_k + s) - f(x_k) \quad \leq \quad -c_{2.1} \| \nabla f(x_k) \| \, \| s \| + \| \nabla f(\xi) - \nabla f(x_k) \| \, \| s \|,$$

where s is the step for which (4.3) holds. Thus

$$c_{2.1} \| \nabla f(x_k) \| \leq \| \nabla f(\xi) - \nabla f(x_k) \|. \tag{4.4}$$

By the integral form of the mean value theorem,

$$\begin{aligned} \| \nabla f(\xi) - \nabla f(x_k) \| &= \left\| \int_0^1 \left[\, \nabla^2 f(x_k + t(\xi - x_k))(\xi - x_k) \, \right] \, dt \right\| \\ &\leq \ \sigma_{\max} \| \xi - x_k \| \quad \leq \quad \sigma_{\max} \, \Delta_k \, \| B \| \, \Gamma^*. \end{aligned}$$

Meanwhile, since $\nabla f(x_*) = 0$, we have

$$\begin{aligned} \| \nabla f(x_k) \| &= \| \nabla f(x_k) - \nabla f(x_*) \| \\ &= \left\| \int_0^1 \left[\, \nabla^2 f(x_* + t(x_k - x_*))(x_k - x_*) \, \right] \, dt \right\| \ \geq \ \sigma_{\min} \| x_k - x_* \|. \end{aligned} \tag{4.5}$$

Combining (4.4) and (4.5) yields

$$c_{2.1} \, \sigma_{\min} \, \| x_k - x_* \| \quad \leq \quad c_{2.1} \, \| \nabla f(x_k) \| \quad \leq \quad \sigma_{\max} \, \| B \| \, \Gamma^* \, \Delta_k.$$

Setting $c_{4.1} = (\sigma_{\max} \| B \| \Gamma^*)/(c_{2.1}\, \sigma_{\min})$ completes the proof. $\square$

Next we have the following elementary result concerning the level sets of f near an isolated local minimizer x_*.

PROPOSITION 4.2. *Under Hypothesis 2, if $x, y \in \mathcal{B}(x_*, \eta)$ and $f(x) \leq f(y)$, then*

$$\| x - x_* \| \leq \kappa^{\frac{1}{2}} \| y - x_* \|, \tag{4.6}$$

where κ is as defined in (4.1).

Proof. Suppose $x, y \in \mathcal{B}(x_*, \eta)$ and $f(x) \leq f(y)$. From Taylor's theorem with remainder and the fact that $\nabla f(x_*) = 0$, we have

$$\begin{aligned} f(y) &= f(x_*) + \frac{1}{2}(y - x_*)^T \nabla^2 f(\xi)(y - x_*) \\ f(x) &= f(x_*) + \frac{1}{2}(x - x_*)^T \nabla^2 f(\omega)(x - x_*) \end{aligned}$$

for ξ and ω on the line segments connecting x_* with y and x, respectively. Since $f(x) \leq f(y)$, we obtain

$$0 \leq f(y) - f(x) = \frac{1}{2}(y - x_*)^T \nabla^2 f(\xi)(y - x_*) - \frac{1}{2}(x - x_*)^T \nabla^2 f(\omega)(x - x_*),$$

whence

$$0 \leq \sigma_{\max} \| y - x_* \|^2 - \sigma_{\min} \| x - x_* \|^2,$$

and thus (4.6). $\square$

We use the previous proposition to show that if we start sufficiently close to x_* with a sufficiently small step-length control parameter Δ_k and we have stopped allowing increases in Δ_k (Hypothesis 1), then pattern search will not move away from a neighborhood of x_*.

PROPOSITION 4.3. *Under Hypotheses 0, 1, and 2, there exist $\delta_{4.3} > 0$ and $\varepsilon_{4.3} > 0$ for which the following holds. For $k \geq N$, where N is as defined in Hypothesis 1, if x_k is an iterate for which $\Delta_k < \delta_{4.3}$ and $\| x_k - x_* \| < \varepsilon_{4.3}$, then for all $\ell \geq k$,*

$$\| x_\ell - x_* \| < \eta,$$

where η is as in (4.2).

Proof. Choose $\delta_{4.3}$ and $\varepsilon_{4.3}$ to satisfy

$$\begin{aligned} \delta_{4.3} &< \frac{\eta}{2c_0} \\ \varepsilon_{4.3} &< \frac{1}{2}\kappa^{-\frac{1}{2}}\eta, \end{aligned}$$

where the constant c_0 comes from Hypothesis 0 and the definition of κ appears as (4.1). Observe that the definition of κ means that for any choice of $\eta > 0$,

$$\frac{1}{2}\kappa^{-\frac{1}{2}}\eta \leq \frac{\eta}{2}.$$

The proof is by induction. First consider $x_{k+1} = x_k + s_k$. Hypothesis 0 gives us $\| x_{k+1} - x_k \| = \| s_k \| \leq c_0 \Delta_k$. We have, *a priori*,

$$\| x_{k+1} - x_* \| \leq \| x_{k+1} - x_k \| + \| x_k - x_* \| < c_0 \Delta_k + \varepsilon_{4.3} < \eta.$$

Now consider any $\ell \geq k+1$, and suppose

$$\| x_\ell - x_* \| < \eta.$$

Then

$$\| x_{\ell+1} - x_* \| \leq \| x_{\ell+1} - x_\ell \| + \| x_\ell - x_* \|. \tag{4.7}$$

Hypothesis 1 assures us that $\Delta_\ell \leq \Delta_k$ for $\ell \geq k$, so

$$\| x_{\ell+1} - x_\ell \| \leq c_0 \Delta_\ell \leq c_0 \Delta_k.$$

Meanwhile, by the induction hypothesis, $x_\ell \in \mathcal{B}(x_*, \eta)$. Since $f(x_\ell) \leq f(x_k)$ as well, Proposition 4.2 and the assumption $\| x_k - x_* \| < \varepsilon_{4.3}$ say that

$$\| x_\ell - x_* \| \leq \kappa^{\frac{1}{2}} \| x_k - x_* \| < \kappa^{\frac{1}{2}} \varepsilon_{4.3}.$$

Thus (4.7) yields

$$\| x_{\ell+1} - x_* \| < c_0 \Delta_k + \kappa^{\frac{1}{2}} \varepsilon_{4.3} < \eta.$$

□

An immediate consequence of Proposition 4.3 is the following, which is simply a localized version of Theorem 3.3 from [16].

PROPOSITION 4.4. *Suppose Hypotheses 0–2 hold. Let $\delta_{4.3} > 0$ and $\varepsilon_{4.3} > 0$ be as in Proposition 4.3. If for some $k \geq N$, where N is as defined in Hypothesis 1, we have $\Delta_k < \delta_{4.3}$ and $\| x_k - x_* \| < \varepsilon_{4.3}$, then $\lim_{j \to \infty} \Delta_j = 0$.*

Proof. The proof proceeds by contradiction. Suppose $\lim_{j \to \infty} \Delta_j \neq 0$. Then Δ_j has some minimum value $\Delta_{\min} > 0$, which implies that after some iteration k we have an infinite number of successful iterations. From Proposition 4.3 and Hypothesis 0, we see that all possible iterates after k remain in a bounded set. As discussed in Section 2.2, the structure of pattern search algorithms is such that all possible iterates must lie on a translated integer lattice that depends on $\Delta_{\min}$. The intersection of a bounded set with a translated integer lattice is finite. So if we do not reduce Δ_j beyond $\Delta_{\min}$, there is only a finite number of points that we can consider that remain in the bounded set. Thus, if there is an infinite number of successful iterations, there must exist at least one point $\hat{x}$ in the lattice for which $x_j = \hat{x}$ for more than one value of j. This leads to a contradiction because we can only have a successful iteration and avoid decreasing Δ_j if $f(x_j) < f(x_{j-1})$. Therefore, we must have $\lim_{j \to \infty} \Delta_j = 0$. □

This argument is analogous to the basic reasoning found in the proof of Theorem 3.3 in [16], in which it is shown that $\liminf_{k \to +\infty} \Delta_k = 0$ under the assumption that $\mathcal{L}(x_0)$ is compact.

Putting the pieces together, we obtain the following local convergence result. It says that if at some iteration the entire set of trial points is sufficiently close to a local minimizer x_* satisfying Hypothesis 2, then the sequence of subsequent iterates will converge to x_*. We use the suggestive notation $x_k + \Delta_k P_k$ to represent the set of all possible trial points at iteration k, $\left\{ x_k + \Delta_k B c_k^i \mid i = 1, \ldots, p_k \right\}$, where B is the basis matrix and c_k^i is a column of the generating matrix C_k.

THEOREM 4.5. *Given a pattern search algorithm satisfying Hypotheses 0–1, let N be as in Hypothesis 1. Suppose Hypothesis 2 holds and that, in particular, x_* is a point satisfying Hypothesis 2.*

Then there exist $\rho > 0$ and $c_{4.5} > 0$ for which the following hold. Suppose that at some iteration K, $K \geq N$, we have $x_K + \Delta_K P_K \subset \mathcal{B}(x_*, \rho)$. Let $\bar{K}$ be the first unsuccessful iteration after K. Then for all $k > \bar{K}$,

$$\| x_k - x_* \| \leq c_{4.5} \Delta_{m(k)}, \tag{4.8}$$

where $m(k)$ is the last unsuccessful iteration preceding or including k. As a consequence, we have $\lim_{k \to \infty} x_k = x_*$.

Proof. We begin by noting that the integrality of the generating matrix C_k guarantees that for all k

$$\min_{i \in \{1, \ldots, (p_k - 1)\}} \| c_k^i \| \geq 1. \tag{4.9}$$

The bound in (4.9) excludes the last column of C_k, which allows the zero step. We also know that for all $k \geq 0$ any trial step $s_k^i \in \Delta_k P_k$ satisfies

$$\| s_k^i \| \; = \; \Delta_k \| B c_k^i \| \; \geq \; \Delta_k \sigma_n(B) \| c_k^i \|, \tag{4.10}$$

where $\sigma_n(B)$ denotes the smallest singular value of the basis matrix B.

Our assumption that $x_K + \Delta_K P_K \subset \mathcal{B}(x_*, \rho)$ means that for any $s_K^i \in \Delta_K P_K$ we have

$$\| s_K^i \| < 2\rho. \tag{4.11}$$

Combining (4.9), (4.10), and (4.11), we obtain

$$\Delta_K < \frac{2\rho}{\sigma_n(B) \| c_K^i \|} \leq \frac{2\rho}{\sigma_n(B)},$$

for all $i \in \{1, \ldots, (p_K - 1)\}$. The assumption that $x_K + \Delta_K P_K \subset \mathcal{B}(x_*, \rho)$ also yields

$$\| x_K - x_* \| < \rho.$$

Thus we can choose $\rho > 0$ to be so small that if $x_K + \Delta_K P_K \subset \mathcal{B}(x_*, \rho)$, then

$$\Delta_K < \min\{\eta, \delta_{4.3}\} \quad \text{and} \quad \| x_K - x_* \| < \min\{\eta, \varepsilon_{4.3}\},$$

where η is as in (4.2) and $\delta_{4.3}$, $\varepsilon_{4.3}$ are as in Proposition 4.3. Proposition 4.3 then gives us

$$\| x_k - x_* \| < \eta \quad \text{for all} \quad k \geq K. \tag{4.12}$$

By assumption, $\bar{K}$ is the first unsuccessful iteration after K. We now consider two cases.

First, for all unsuccessful iterates x_k with $k \geq \bar{K}$, Proposition 4.1 gives us

$$\| x_k - x_* \| \leq c_{4.1} \Delta_k. \tag{4.13}$$

Since x_k is an unsuccessful iterate and Δ_k has not yet been reduced, $k = m(k)$; and we can restate (4.13) as

$$\| x_{m(k)} - x_* \| \leq c_{4.1} \Delta_{m(k)} \quad \text{for all} \quad k \geq \bar{K}. \tag{4.14}$$

Second, for all successful iterations $k > \bar{K}$, we have $f(x_k) < f(x_{m(k)})$. Since $k > \bar{K} \geq K$, (4.12) assures us that $x_k, x_{m(k)} \in \mathcal{B}(x_*, \eta)$. It then follows from Proposition 4.2 that

$$\| x_k - x_* \| \leq \kappa^{\frac{1}{2}} \| x_{m(k)} - x_* \|, \tag{4.15}$$

where κ is as in (4.1).

Together (4.14) and (4.15) imply that for all $k > \bar{K}$, (4.8) holds with $c_{4.5} = \kappa^{\frac{1}{2}} c_{4.1}$ since the definition of κ in (4.1) ensures that $\kappa^{\frac{1}{2}} \geq 1$.

Finally, since Proposition 4.4 says $\Delta_k \to 0$, it follows that $\lim_{k\to\infty} x_k = x_*$. $\square$

This theorem complements Theorem 3.7 of [16], where it is shown, under different hypotheses and a more stringent criterion for accepting a step, that $\| \nabla f(x_k) \| \to 0$. The trade-off is that while here we relax the criterion for accepting a step, Hypothesis 2 places stronger assumptions on f than those used in [16], where all that is assumed about f is that it is continuously differentiable on a neighborhood of $\mathcal{L}(x_0)$.

Theorem 4.5 is similar to local convergence results for other minimization algorithms. The standard convergence results for Newton's method and quasi-Newton methods (with exact gradients) say that if we start sufficiently close to a point x_* satisfying Hypothesis 2, then the sequence of subsequent iterates will converge to x_* [15]. Our result is even more like the local convergence results for minimization with finite-difference estimates of the gradient, with which pattern search can be aptly compared. Theorem 5.1 in [3], an example of such a result, requires the points from whose objective values the finite-difference estimate of the gradients is constructed to lie sufficiently close to x_*. Our requirement that the entire pattern be close to x_* is similar.

Theorem 4.5 says that for the subsequence of *unsuccessful* iterates, the rate of convergence is R-linear. Theorem 4.5 says nothing about what may happen at the successful iterations, nor how many such iterations there may be between unsuccessful iterations. The obstruction to sharpening the rate of convergence result is that we do not know *a priori* how much improvement we obtain in $f(x)$ at the successful iterations. We have a sort of multi-step R-linear rate of convergence, but one for which we do not know and, as our numerical tests reported in Section 5.3 suggest, cannot predict, the number of intervening steps. For want of an existing term for this notion of convergence, we call it *desultory R-linear convergence*.

More positively, Theorem 4.5 suggests how one can "accelerate" the local convergence of pattern search algorithms. One need only rename the formerly unsuccessful iterates successful iterates and drop the formerly successful iterates from discussion. Then, *mirabile dictu*, this simple modification makes the successful iterates an R-linearly convergent sequence.

All joking aside, this suggestion is based on the observation that we can rewrite pattern search algorithms to have an inner iteration/outer iteration structure. The outer iterations consist of those iterates at which we reduce Δ_k because no more local reduction in f can be found using the current pattern $\Delta_k P_k$. The inner iterations comprise those iterates which identify simple decrease for some $s_k \in \Delta_k P_k$. Theorem 4.5 allows us to say something about the asymptotic behavior of such "outer" iterations in pattern search.

In that sense, our results are similar to the local convergence for, say, steepest descent with a line search strategy. In steepest descent, the line search is an inner iteration that may require multiple evaluations of the objective in order to generate the ostensible next iterate. In this way both pattern search and steepest descent generate R-linearly convergent sequences. However, we do not see a way to say anything, asymptotically, about the behavior of the pattern search "inner" iterations. By contrast, one can bound, asymptotically, the number of steps required for the inner iterations of steepest descent devoted to the exercise of a linesearch strategy, since in the worst case one builds a quadratic model of the objective along the search direction. Once again, for the pattern search analysis the gap lies in the lack of both an explicit estimate of the gradient and a local model of f with which to work. Faster local convergence seems to require better local models.

We close by noting that the only other local convergence result for pattern search similar to Theorem 4.5 of which we are aware is due to Yu [18]. The result is restricted to positive definite quadratic functions (though the extension to nonlinear objectives is straightforward). The fact that f is a quadratic figures explicitly in the derivation of a result similar to (4.8).

5. Numerical results. We now present some numerical experiments that illustrate the practical implications of our convergence analysis. The first round of testing, summarized in Section 5.2, supports the analysis; the second round, summarized in Section 5.3, shows its limitations. The numerical results regarding the effectiveness of Δ_k as a measure of stationarity, reported in Section 5.2, summarize some of the numerical results reported in [6]. The second round of results, given in Section 5.3, was generated using the implementation of pattern search from [6].

5.1. The testing environment. Full details of the numerical experiments can be found in [6]. The tests we report here were done with randomly generated quadratic functions. This is a reasonable choice, since we are interested in the local convergence behavior of pattern search, and any function that is twice continuously differentiable looks like a convex quadratic in the neighborhood of an isolated local minimizer. The quadratics tested were of the form $f(x) = x^T A x + c$, where $A = H^T H$ and $H \in \mathbb{R}^{(n+2)\times n}$ is a matrix with entries that are normal random variates with means of zero and standard deviations of one. The absence of a linear term may be thought of as shifting the quadratic so that the solution lies at the origin, which simplifies our calculations of $\| x_k - x_* \|$. The constant term c is not interesting for the purposes of the optimization but provides a useful tag for identifying individual functions. For the testing in [6], $2 \leq n \leq 5$; we show a result for $n = 5$.

In addition to randomly generating the entries of the matrix H, we also randomly generated Δ_0 and the entries of the vector x_0. The entries for the starting point x_0 were also normal random variates with means of zero and standard deviations of one. The choice for Δ_0 was an exponential variate with a mean of one. Since the starting points are randomly generated, the absence of a linear term in the quadratic should not unduly influence the outcome of the searches.

The software described in [6] was written in C++ to make use of C++ classes, a convenient way to establish the key features of pattern search and then easily derive specific variants. Several of these variants were implemented and tested, as described in [6]. We show results using `HJSearch`, an implementation of the classical pattern search algorithm of Hooke and Jeeves [8]; `CompassSearch`, the pattern search algorithm described in Section 2; and `NLessSearch`, a pattern search algorithm that takes advantage of the fact that a minimal positive basis requires only $n+1$ vectors [11], as opposed to the $2n$ coordinate vectors used in most traditional pattern search methods, including compass search and Hooke and Jeeves.

5.2. Measuring stationarity. The first question we ask is: how effective is Δ_k as a measure of stationarity? Not too surprisingly, the results of our tests showed that Δ_k is a reliable measure of progress toward a solution. Furthermore, our numbers make quite clear the R-linear convergence of the subsequence of unsuccessful iterates.

After any unsuccessful iteration, a pattern search method is required to reduce Δ_k. We used the standard reduction factor of $\frac{1}{2}$ so that after an unsuccessful iteration, $\Delta_{k+1} = \frac{1}{2}\Delta_k$. Before proceeding to the next iteration, we recorded the value of Δ_k, $\| \nabla f(x_k) \|$, $| f(x_k) - f(x_*) |$, and $\| x_k - x_* \|$ (though since we knew $x_* \equiv 0$, we simply had to compute $\| x_k \|$). Representative results from one particular test are given in Table 5.1.

Table 5.1
`HJSearch` *in five variables*

Δ_k	$\| \nabla f(x_k) \|$	$\| f(x_k) - f(x_*) \|$	$\| x_k - x_* \|$
0.696226813823902	3.718628968450993	3.96639084353257	2.396301558944381
0.348113406911951	1.370661155865317	0.44618879006458	0.698389592313846
0.174056703455976	0.993386770046628	0.19091214014793	0.450386903073632
0.087028351727988	0.236893510661273	0.01477525409286	0.153943970082610
0.043514175863994	0.314026005456998	0.01421309666224	0.119315177505950
0.021757087931997	0.131650296045321	0.00223337373949	0.034002609804365
0.010878543965999	0.042526372212693	0.00028996796577	0.015791910616849
0.005439271982999	0.032921235371376	0.00018078437086	0.014678346820778
0.002719635991500	0.012854930063180	0.00014567060113	0.016582990396810
0.001359817995750	0.005667414556147	0.00001023084696	0.003757596625046
0.000679908997875	0.004101406209192	0.00000429756349	0.002612852810391
0.000339954498938	0.001396318029208	0.00000050775161	0.000854609846084
0.000169977249469	0.000833651146770	0.00000049903818	0.000985750547712
0.000084988624734	0.000563050121378	0.00000002356890	0.000074244150774
0.000042494312367	0.000112117511534	0.00000000325088	0.000043510325021
0.000021247156184	0.000097664692564	0.00000000266601	0.000032689236837
0.000010623578092	0.000035578092711	0.00000000026108	0.000013878637584
0.000005311789046	0.000010624256256	0.00000000015362	0.000017458315183

The point of the results we report in Table 5.1 is not to demand close scrutiny of each entry but rather to demonstrate the trends in each of the four quantities measured. We clearly see the R-linear behavior the analysis tells us to expect: by the time we halve Δ_k, we have roughly halved the error in the solution.

We report here the results from only one experiment, but they are representative of results from ten thousand runs over multiple quadratics, in multiple dimensions, from multiple starting points, with multiple choices of Δ_0, using four different pattern search methods [6]. We found that across all these tests, Δ_k gave us a consistent measure of the accuracy of the solution. Further, these results conform both with a long-standing recommendation for a stopping criterion (see [8]) as well as with our observations when applying pattern search algorithms to general (i.e., non-quadratic) functions.

One practical benefit of using Δ_k as a measure of stationarity is that it is already present in pattern search algorithms; no additional computation is required. Another good reason for using Δ_k as a measure of stationarity is that it is largely insusceptible to numerical error. Since pattern search methods often are recommended when the evaluations of the objective function are subject to numerical "noise," the fact that Δ_k will not be affected by numerical noise in the computed values of the objective function suggests that Δ_k provides a particularly suitable stopping criterion. One last observation to be made about the practical utility of Δ_k as a measure of stationarity is that pattern searches only requires ranking, or order, information to drive the search—no numeric values for the objective are necessary [11]. In such a setting, Δ_k is a feasible measure of progress whereas measures based on the numeric values of the objective function are not.

We close with the observation that the conditioning of the Hessian does play a role in the progress of the search, as is true for steepest descent. For the example in Table 5.1 this is not an issue since the smallest singular value for the Hessian is 0.4661 and the condition number of the Hessian is 37.5767. However, in limited tests, we parameterized the Hessian of a two-dimensional quadratic to control the condition number of A. As the Hessian became increasingly less well-conditioned, the number of iterations between each unsuccessful iteration grew; however, we still saw the same trends evident in Table 5.1. The effect the conditioning of the Hessian has on our experimental results should not be surprising since the constant $c_{4.1}$ in Proposition 4.1 explicitly depends on $\sigma_{\min}$; as $\sigma_{\min} \to 0$, $c_{4.1} \to \infty$. For a similar observation regarding the connection between conditioning and the performance of steepest descent with finite-difference gradients, see [3].

5.3. How many successful iterates? Theorem 4.5 says that the subsequence of unsuccessful iterates converges R-linearly once we are in a neighborhood of a solution. A natural question to then ask is: how many iterations occur in practice between each iterate included in this subsequence? If a reasonable *a priori* bound for the number of intervening iterations could be derived, then we could establish the rate of convergence for the entire sequence of iterates. Since we could see no analytical approach to answering this question, as discussed at the end of Section 4, we decided to conduct some numerical studies. As it happens, our experiments shed little light on the question. We give only a few specific results in Figures 5.1–5.2.

In all instances, we terminate the search when $\Delta_{k+1} < 2 \times 10^{-8}$. Along the horizontal axis, we list the number of unsuccessful iterations; i.e., the number of times we halve Δ_k before it is less than the stopping tolerance. Each bar then represents the number of successful iterations that preceded an unsuccessful iteration plus the (single) unsuccessful iteration so that summing all the entries gives us the total number of iterations for the search.

Notice that for the three algorithms we tested the scale on the vertical axes varies considerably. For `NLessSearch`, the number of successful iterations preceding an unsuccessful iteration can be considerably higher than, say, for `HJSearch`, but over all of our tests, the results are mixed. We cannot predict how many successful iterations may precede an unsuccessful iteration, nor does there seem to be any particular trend. However, a few useful observations emerged.

One trend that can be seen in Figures 5.1–5.2 is the apparent superiority, in terms of the total number of iterations required to satisfy our stopping criterion, of the algorithm of Hooke and Jeeves when applied to quadratic functions. This is consistent with the results in [6]. As yet we can offer no analytical explanation for this behavior, but it seems that the "pattern step" in the Hooke and Jeeves algorithm, which captures some limited history of prior successes and potentially enables a much longer trial step than allowed by the core pattern, helps the overall progress of the search.

Another point is illustrated by the example shown in Figure 5.2. The poor scaling of the graphs in Figure 5.2, a consequence of the relatively huge number of iterations

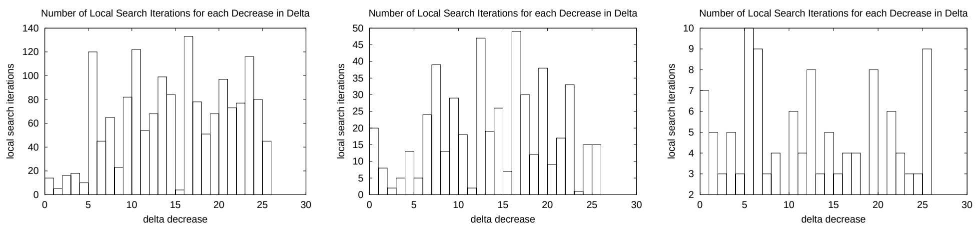

FIG. 5.1. `NLessSearch` *(left),* `CompassSearch` *(middle), and* `HJSearch` *(right) in 8 variables.*

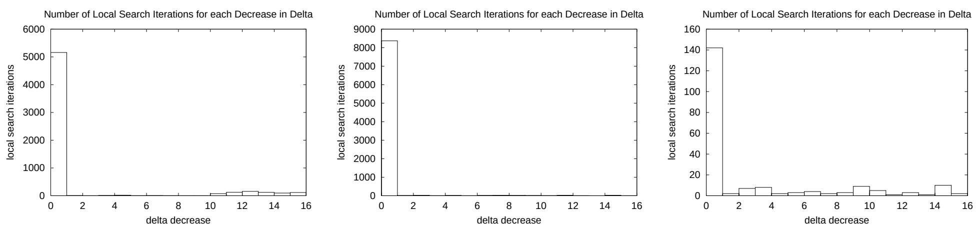

FIG. 5.2. `NLessSearch` *(left),* `CompassSearch` *(middle), and* `HJSearch` *(right) in 4 variables.*

taken before the first reduction in Δ, precludes close examination—but that underscores the point we wish to make.

The relatively huge number of successful iterations before Δ_k is ever reduced is due to the small initial value of Δ_0. For our experiments, the value of Δ_0 was drawn randomly. In this example it is so small (0.001128116614106) that initially there is a long sequence of successful iterations, but progress is remarkably slow because we start with such a small choice of Δ_0 that all the trial steps are quite short. After the first reduction in Δ, the number of iterations between each subsequent reduction in Δ demonstrates the same unpredictable behavior we see in the graphs in Figure 5.1.

This suggests two conjectures. The first is that in general it is best to start the search with a relatively large value of Δ_0. This is consistent with pattern search/direct search lore (e.g., see the discussion found in [17] on choosing the size of the initial simplex). The second conjecture is that there is merit to allowing Δ_k to increase so as to recover from an inappropriate choice of Δ_0. While the analyses in [16, 11] support such a specification for pattern search algorithms, most analyses require Δ_k to be monotonically nonincreasing. Furthermore, we are aware of only two publicly-available implementations of pattern search methods [7, 9] that allow Δ_k to increase. Even the analysis we present here assumes that eventually Δ_k is monotonically nonincreasing. The practical compromise, implicit in Hypothesis 1, is that we allow increases in Δ_k only up to some finite number of iterations, after which we require Δ_k to be nonincreasing. This allows for some initial adjustments in the step-length control parameter if the first few iterations of the search suggest that the choice of Δ_0 may have been too conservative. However, if we disable any further increases in Δ_k once $k \geq N$, then we preserve the global and local convergence properties presented in Sections 3 and 4.

6. Conclusion. The results given here round out the convergence analysis of pattern search. The analysis and numerical experiments reported here show that Δ_k can be used as a reliable stopping criterion. Moreover, these tests show that the correlations predicted by Theorems 3.1 and 4.5 between Δ_k, $\|\nabla f(x_k)\|$, and $\|x_k - x_*\|$

are manifest in practice. These results vindicate the intuition of the early developers of direct search methods.

Acknowledgments. We are indebted to Natalia Alexandrov for a conversation that led to the term "desultory convergence" in connection with Theorem 4.5. We thank Stephen Nash for a lively discussion about stopping criteria.

We also thank both referees and the associate editor for their careful reading of the earlier drafts of this paper. They caught an oversight in one of the proofs and made many helpful suggestions for improving the overall presentation. We greatly appreciate their efforts.

REFERENCES

[1] C. Audet, *Convergence results for pattern search algorithms are tight*, Tech. Rep. 98–24, Department of Computational and Applied Mathematics, Rice University, 6100 Main Street, MS 134, Houston, Texas 77005–1892, 1998.

[2] M. Avriel, *Nonlinear Programming: Analysis and Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1976.

[3] P. T. Boggs and J. J. E. Dennis, *A stability analysis for perturbed nonlinear iterative methods*, Mathematics of Computation, 30 (1976), pp. 199–215.

[4] M. J. Box, D. Davies, and W. H. Swann, *Non-Linear Optimization Techniques*, ICI Monograph No. 5, Oliver & Boyd, Edinburgh, 1969.

[5] C. Davis, *Theory of positive linear dependence*, American Journal of Mathematics, 76 (1954), pp. 733–746.

[6] E. D. Dolan, *Pattern search behavior in nonlinear optimization.* Honors Thesis, Department of Computer Science, College of William & Mary, Williamsburg, Virginia 23187–8795, May 1999. Accepted with Highest Honors. Available at `http://www.cs.wm.edu/~va/CS495/`.

[7] A. P. Gurson, *Simplex search behavior in nonlinear optimization.* Honors Thesis, Department of Computer Science, College of William & Mary, Williamsburg, Virginia 23187–8795, May 2000. Accepted with Highest Honors. Available at `http://www.cs.wm.edu/~va/CS495/`.

[8] R. Hooke and T. A. Jeeves, *Direct search solution of numerical and statistical problems*, Journal of the Association for Computing Machinery, 8 (1961), pp. 212–229.

[9] P. D. Hough, T. G. Kolda, and V. J. Torczon, *Asynchronous parallel pattern search for nonlinear optimization*, SIAM Journal on Scientific Computing, 23 (2001), pp. 134–156.

[10] T. G. Kolda and V. J. Torczon, *On the convergence of asynchronous parallel pattern search*, Tech. Rep. SAND2001–8696, Sandia National Laboratories, Livermore, California, February 2002. Revised July 2002 for *SIAM Journal on Optimization*.

[11] R. M. Lewis and V. J. Torczon, *Rank ordering and positive bases in pattern search algorithms*, Tech. Rep. 96–71, Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, Virginia 23681–2199, 1996.

[12] ———, *Pattern search algorithms for bound constrained minimization*, SIAM Journal on Optimization, 9 (1999), pp. 1082–1099.

[13] ———, *Pattern search methods for linearly constrained minimization*, SIAM Journal on Optimization, 10 (2000), pp. 917–941.

[14] ———, *A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds*, SIAM Journal on Optimization, 12 (2002), pp. 1075–1089.

[15] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw–Hill, New York, 1996.

[16] V. Torczon, *On the convergence of pattern search algorithms*, SIAM Journal on Optimization, 7 (1997), pp. 1–25.

[17] F. H. Walters, L. R. Parker, Jr., S. L. Morgan, and S. N. Deming, *Sequential Simplex Optimization*, Chemometrics Series, CRC Press, Inc., Boca Raton, Florida, 1991.

[18] W. Yu, *Positive basis and a class of direct search techniques*, Scientia Sinica, Special Issue of Mathematics, 1 (1979), pp. 53–67.