SIAM J. OPTIM.
Vol. 9, No. 4, pp. 1082–1099

PATTERN SEARCH ALGORITHMS FOR BOUND CONSTRAINED MINIMIZATION*

ROBERT MICHAEL LEWIS† AND VIRGINIA TORCZON‡

To John Dennis, on his 60th birthday

Abstract. We present a convergence theory for pattern search methods for solving bound constrained nonlinear programs. The analysis relies on the abstract structure of pattern search methods and an understanding of how the pattern interacts with the bound constraints. This analysis makes it possible to develop pattern search methods for bound constrained problems while only slightly restricting the flexibility present in pattern search methods for unconstrained problems. We prove global convergence despite the fact that pattern search methods do not have explicit information concerning the gradient and its projection onto the feasible region and consequently are unable to enforce explicitly a notion of sufficient feasible decrease.

Key words. bound constrained optimization, convergence analysis, pattern search methods, direct search methods, globalization strategies, alternating variable search, axial relaxation, local variation, coordinate search, evolutionary operation, multidirectional search

AMS subject classifications. 49M30, 65K05

PII. S1052623496300507

1. Introduction.

This paper extends the class of pattern search methods for unconstrained minimization, considered in [16], to bound constrained problems:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & \ell \leq x \leq u, \end{array} \tag{1.1}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $\ell, x, u \in \mathbf{R}^n$, and $\ell < u$. We allow the possibility that some of the variables are unbounded either above or below by permitting $\ell_j, u_j = \pm\infty$, $j = 1, \ldots, n$.

Our convergence analysis is guided by that for pattern search methods for unconstrained problems [16]. We can guarantee that if the objective f is continuously differentiable, then a subsequence of the iterates produced by a pattern search method for problems with bound constraints converges to a stationary point of problem (1.1). By a stationary point of problem (1.1) we mean a feasible point x that satisfies the first-order necessary condition for optimality: for all feasible z, $(\nabla f(x)\ ,\ z - x) \geq 0$. Equivalently, x is a Karush–Kuhn–Tucker point for problem (1.1). As in the case of unconstrained minimization, pattern search methods for bound constrained problems accomplish this without an explicit representation of the gradient or the directional derivative. In particular, we prove global convergence in the bound constrained case

*Received by the editors March 13, 1996; accepted for publication (in revised form) January 12, 1998; published electronically September 24, 1999.
http://www.siam.org/journals/siopt/9-4/30050.html

†Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, VA 23681-2199 (buckaroo@icase.edu). This research was supported by the National Aeronautics and Space Administration under NASA contract NAS1-19480.

‡Department of Computer Science, College of William & Mary, Williamsburg, VA 23187–8795 (va@cs.wm.edu). This research was supported by Air Force Office of Scientific Research grant F49620–95–1–0210, Sandia National Laboratories, Lawrence Livermore RFQ LC-2683, the National Aeronautics and Space Administration under NASA contract NAS1-19480 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA, and by the Center for Research on Parallel Computation through NSF Cooperative Agreement 9120008.

even though pattern search methods do not have explicit information concerning the gradient and its projection onto the feasible region and consequently do not explicitly enforce a notion of sufficient feasible decrease.

In (1.1), near the boundary of the feasible region, the proximity of the boundary restricts the set of descent directions along which we can search *and* remain feasible for a sufficiently long distance. In projected gradient methods, one circumvents this inconvenience by combining knowledge about the local behavior of the objective f, namely, the gradient, with the global structure of the feasible region by conducting searches along the projected gradient path. In the case of pattern search methods, we do not have recourse to this strategy; nonetheless, we can specify the pattern so that it contains a sufficiently rich set of directions to ensure that we need not take too short a step to obtain a new iterate that produces decrease in f and is also feasible.

So far as we know, ours is the first convergence analysis for pattern search methods for bound constrained minimization. However, the observation that forms the basis of our analysis—the utility of having a sufficiently large subset of the pattern oriented along the coordinate directions in order to handle the bounds—is not new. For instance, in [10], Keefer notes that the pattern associated with the method of Hooke and Jeeves [9] is well suited for coping with bounds and proposes the Simpat algorithm, which combines the use of the Nelder–Mead simplex algorithm [12] in the interior of the feasible region with the use of the Hooke and Jeeves pattern search algorithm near the boundary.

The general specification of pattern search methods for bound constrained minimization gives us broad latitude in designing such algorithms. Moreover, as we shall discuss, classical pattern search methods for unconstrained minimization—such as coordinate search with fixed step sizes and the original pattern search of Hooke and Jeeves—can be generalized without modification to the bound constrained case. We also will show that not all pattern search methods for unconstrained minimization immediately generalize to bound constrained problems: in section 2 we present a counterexample that defeats G.E.P. Box's method of evolutionary operation using two-level factorial designs [1, 3, 14] and show how the convergence theory guides us to a remedy that uses composite designs [2], instead of the simpler factorial or fractional factorial designs. The multidirectional search algorithm of Dennis and Torczon [7, 15] also requires us to augment the pattern used for the algorithm; again we find a straightforward extension, but one that reveals much about the interesting behavior of the simplices which characterize that method.

2. Motivation. Before giving the technical specification of pattern search methods for bound constrained minimization, we consider an example that illustrates what is needed for the generalization and how the bound constrained algorithms work. Consider the following simple linear problem:

$$\begin{array}{ll} \text{minimize} & f(x) = -(x^1 + 2x^2) \\ \text{subject to} & \quad 0 \leq x^1 \leq 1, \\ & \qquad x^2 \leq 0. \end{array}$$

The solution of this problem is $x_* = (1, 0)^T$. Let us consider an iteration of the pattern search method of evolutionary operation applied to this problem starting at the initial iterate $x_0 = (0, 0)^T$.

The usual pattern is typically a factorial design comprising the points NW, NE, SW, and SE indicated by the open circles in Figure 2.1. We see that the values of f at the points NW and NE are lower than that at x_0. If there were no constraints, as

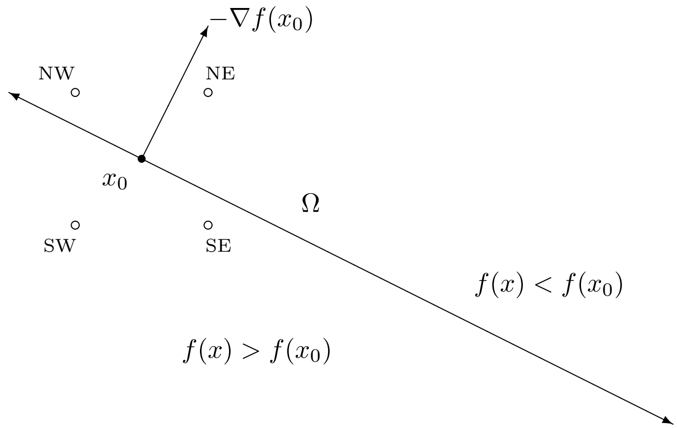

FIG. 2.1. *The pattern for factorial design in the unconstrained case.*

depicted in Figure 2.1, the algorithm could choose either of these points as the next iterate; most implementations would choose NE since it produces the greater decrease in f.

In the unconstrained case, pattern search methods work much like line-search quasi-Newton methods. Pattern search methods include sufficient search directions to guarantee that if the current iterate is not a stationary point, then at least one of the search directions is a descent direction. Moreover, one can prove that as the iterations progress, these "good" search directions cannot become increasingly orthogonal to the steepest descent direction. In the situation depicted above, for instance, regardless of the direction of $-\nabla f(x_0)$, one of the four directions from x_0 to the corners of the square the pattern defines must make an angle of $45°$ or less with $-\nabla f(x_0)$. Finally, the way the pattern is rescaled implements a form of backtracking that is the final piece needed to guarantee convergence.

Now consider what happens in our simple example when we take into account the constraints. We will consider only feasible points in the pattern, in order to ensure that the algorithm produces only feasible iterates. In Figure 2.2 we see that the only feasible point is SE. Unfortunately, this step will produce increase in f. We cannot remedy this by moving the pattern closer to x_0—backtracking along the directions from x_0 to the points in the pattern—since the only feasible points that will ensue lie along the line segment from x_0 to SE, and on this line segment f is larger than $f(x_0)$. Consequently, evolutionary operation will never move from x_0.

The problem is that while there are feasible directions of descent emanating from x_0, our pattern is not oriented in such a way as to capture any of this information from its feasible point SE. The pattern associated with evolutionary operation is not compatible with the geometry of the feasible region. A moment's reflection reveals that the problem is that the pattern does not allow us to move parallel to the bounds.

This problem goes away if, for instance, we augment the pattern using the idea of composite design [2] (as opposed to factorial design). An example of such a design is shown in Figure 2.3. We now have a feasible step along the active constraint $x^2 \leq 0$ that will produce descent.

This simple example captures the essential idea for the generalization of pattern search methods to bound constrained minimization. We restrict our attention to

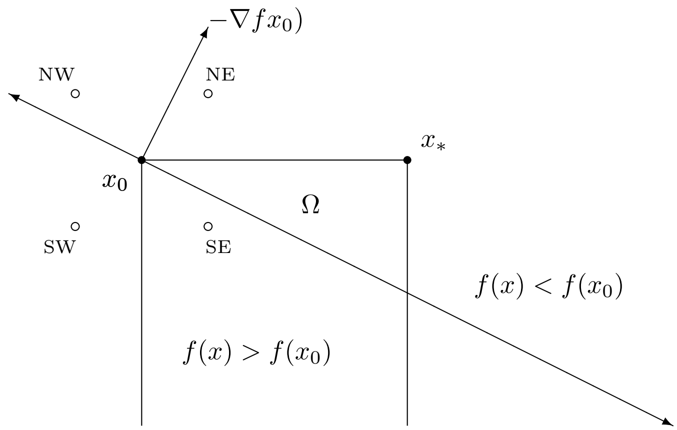

FIG. 2.2. *An illustration of what can go wrong with factorial design in the bound constrained case.*

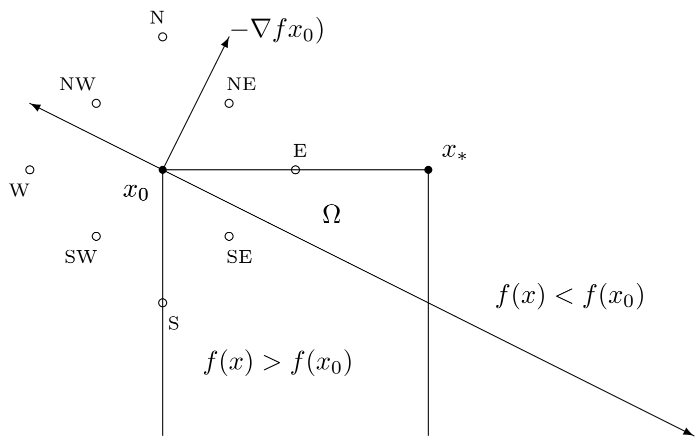

FIG. 2.3. *An illustration of how the problem can be circumvented using a composite design.*

patterns that reflect the geometry of the feasible region by including enough directions oriented along the coordinate axes so that we can move parallel and perpendicular to the boundary of the feasible region. We can then guarantee global convergence to a Karush–Kuhn–Tucker point.

Notation. We denote by $\mathbf{R}$, $\mathbf{Q}$, $\mathbf{Z}$, and $\mathbf{N}$ the sets of real, rational, integer, and natural numbers, respectively.

Unless otherwise noted, norms are assumed to be the Euclidean norm. The feasible region for problem (1.1) we denote by Ω:

$$\Omega = \{ \, x \in \mathbf{R}^n \;\mid\; \ell \leq x \leq u \, \} \, .$$

The projection onto Ω we denote by P. If for scalar t we define

$$p_j(t) = \begin{cases} \ell_j & \text{if } t < \ell_j, \\ t & \text{if } \ell_j \leq t \leq u_j, \\ u_j & \text{if } t > u_j, \end{cases}$$

then the projection of $x = (x_1, \ldots, x_n)^T$ is given by

$$P(x) = \sum_{j=1}^{n} p_j(x_j) e_j,$$

where $\{e_j\}$, $j = 1, \ldots, n$, are the standard basis vectors. On those few occasions where we must denote components of subscripted vectors, we use the following notation: $q_{k,j}$ denotes the jth component of the vector q_k.

We will denote by $g(x)$ the gradient $\nabla f(x)$ of the objective. Finally, let

$$L_\Omega(y) = \{ \ x \in \Omega \ \mid \ f(x) \leq f(y) \ \}.$$

3. Pattern search methods. We begin by defining the general pattern search method for the bound constrained problem (1.1); it differs from that for unconstrained problems [16] in only a few particulars, which we summarize in section 3.5.

3.1. The pattern. As with pattern search methods for unconstrained problems, to define a pattern we need two components: a *basis matrix* and a *generating matrix*.

The basis matrix is a nonsingular matrix $B \in \mathbf{R}^{n \times n}$.

The generating matrix is a matrix $C_k \in \mathbf{Z}^{n \times p}$, where $p > 2n$. We partition the generating matrix into components

$$C_k = [\ M_k \quad -M_k \quad L_k \ ] = [\ \Gamma_k \quad L_k \ ]. \tag{3.1}$$

We require that $M_k \in \mathbf{M} \subset \mathbf{Z}^{n \times n}$, where $\mathbf{M}$ is a finite set of nonsingular matrices, and that $L_k \in \mathbf{Z}^{n \times (p-2n)}$ and contains at least one column, a column of zeros.

A *pattern* P_k is then defined by the columns of the matrix $P_k = BC_k$. For convenience, we use the partition of the generating matrix C_k given in (3.1) to partition P_k as follows:

$$P_k = BC_k = [\ BM_k \quad -BM_k \quad BL_k \ ] = [\ B\Gamma_k \quad BL_k \ ].$$

We also require the matrix BM_k to be diagonal:

$$BM_k = \mathrm{diag}(d_k^i), \quad i = 1, \ldots, n. \tag{3.2}$$

This condition, absent in the case of unconstrained minimization, is needed in order to ensure that we can find feasible points in the pattern that will also produce decrease in the objective. As we shall see, this condition is not especially restrictive and is satisfied by all of the commonly encountered pattern search algorithms or straightforward variants of them.

At iteration k, given $\Delta_k \in \mathbf{R}$ with $\Delta_k > 0$, we define a *trial step* to be a vector of the form $s_k^i = \Delta_k B c_k^i$ for some $i \in \{1, \ldots, p\}$, where c_k^i denotes the ith column of C_k (i.e., $C_k = [c_k^1 \cdots c_k^p]$). We call a trial step s_k^i *feasible* if $(x_k + s_k^i) \in \Omega$. At iteration k, a *trial point* is any point of the form $x_k^i = x_k + s_k^i$, where x_k is the current iterate.

3.2. The bound constrained exploratory moves. Pattern search methods proceed by conducting a series of *exploratory moves* about the current iterate x_k to choose a new iterate $x_{k+1} = x_k + s_k$ for some feasible step s_k determined during the course of the exploratory moves. The hypotheses listed in Figure 3.1 on the result of the bound constrained exploratory moves allow a broad choice of exploratory moves while ensuring the properties required to prove convergence. By abuse of notation, if A is a matrix, $y \in A$ means that the vector y is a column of A.

1. $s_k \in \Delta_k P_k \equiv \Delta_k B C_k \equiv \Delta_k \, [B\Gamma_k \;\; BL_k]$.
2. $(x_k + s_k) \in \Omega$.
3. If $\min\{\, f(x_k + y) \;\mid\; y \in \Delta_k B\Gamma_k,\; x_k + y \in \Omega \,\} < f(x_k)$, then $f(x_k + s_k) < f(x_k)$.

FIG. 3.1. *Hypotheses on the result of the bound constrained exploratory moves.*

Let $x_0 \in \Omega$ and $\Delta_0 > 0$ be given.
For $k = 0, 1, \ldots,$

a. compute $f(x_k)$.
b. determine a step s_k using a bound constrained exploratory moves algorithm.
c. if $f(x_k + s_k) < f(x_k)$, then $x_{k+1} = x_k + s_k$. Otherwise $x_{k+1} = x_k$.
d. update C_k and Δ_k.

FIG. 3.2. *The generalized pattern search method for bound constrained problems.*

Let $\tau \in \mathbf{Q}$, $\tau > 1$, and $\{w_0, w_1, \ldots, w_L\} \subset \mathbf{Z}$, $w_0 < 0$, and $w_i \geq 0$, $i = 1, \ldots, L$. Let $\theta = \tau^{w_0}$ and $\lambda_k \in \Lambda = \{\tau^{w_1}, \ldots, \tau^{w_L}\}$.

a. If $f(x_k + s_k) \geq f(x_k)$, then $\Delta_{k+1} = \theta \Delta_k$.
b. If $f(x_k + s_k) < f(x_k)$, then $\Delta_{k+1} = \lambda_k \Delta_k$.

FIG. 3.3. *Rules for updating Δ_k.*

3.3. The generalized pattern search method. Figure 3.2 states the generalized pattern search method for minimization with bound constraints. To define a particular pattern search method, we must specify the basis matrix B, the generating matrix C_k, the bound constrained exploratory moves to be used to produce a feasible step s_k, and the algorithms for updating C_k and Δ_k.

3.4. The updates. Figure 3.3 specifies the rules for updating Δ_k. The aim of the update of Δ_k is to force a strict reduction in f. An iteration with $f(x_k + s_k) < f(x_k)$ is *successful;* otherwise, the iteration is *unsuccessful*. Note that to accept a step we require only *simple*, as opposed to *sufficient*, decrease.

The conditions on θ and Λ ensure that $0 < \theta < 1$ and $\lambda_i \geq 1$ for all $\lambda_i \in \Lambda$. Thus, if an iteration is successful it may be possible to increase the step length parameter Δ_k, but Δ_k is not allowed to decrease.

3.5. Differences between pattern search methods for unconstrained and bound constrained minimization. There are only two additional restrictions required of pattern search methods to ensure convergence for the bound constrained case.

First note that as we have defined them, pattern search methods for bound constrained minimization are *feasible point* methods; the search begins with a point that satisfies the bounds and maintains feasibility throughout the search. This can be seen in Figure 3.2, where we require $x_0 \in \Omega$. This requirement also appears in the hypotheses on the result of the bound constrained exploratory moves given in Figure 3.1: if simple decrease on the function value at the current iterate can be found among any of the feasible trial steps contained in the columns of $\Delta_k B\Gamma_k$, then the exploratory moves must produce a feasible step s_k that also gives simple decrease on the function value at the current iterate.

The second, and more interesting, restriction is that the *core pattern* BM_k must be defined by a diagonal matrix. Because the columns of the pattern matrix determine the directions of the steps that may be considered, we need to ensure that if we are not at a constrained stationary point, we have at least one feasible direction of descent. Moreover, we need a feasible direction of descent along which we will remain feasible for a sufficiently long distance to avoid taking too short a step. This is a crucial point since we do not enforce any notion of sufficient decrease. Practically, we must ensure that we have directions that allow us to move parallel to the constraints. Requiring BM_k to be a diagonal matrix is sufficient, and as we saw in section 2, such a requirement is unavoidable.

We note an equivalence between pattern search methods for bound constrained problems and an exact penalization approach to problem (1.1). Applying a pattern search method for problem (1.1) produces exactly the same iterates as applying such an algorithm to the unconstrained problem

$$\text{minimize} \quad F(x),$$

where

$$F(x) = \begin{cases} f(x) & \text{if } x \in \Omega, \\ \infty & \text{otherwise.} \end{cases}$$

In fact, this is one classical approach used with direct search methods to ensure that the iterates produced remain feasible (see, for instance, [10, 12, 13]). In the case of pattern search methods this formulation is not simply a conceptual approach; pattern search methods are directly applicable to this exact penalty function since they do not rely on derivatives. However, as we demonstrated in section 2, this exact penalization approach cannot be applied with an arbitrary pattern search method for unconstrained minimization; we require that BM_k be diagonal.

3.6. Results from the unconstrained theory. We recall the following results from [16], to which we refer the reader for the proofs. The first result indicates one sense in which Δ_k regulates step length.

LEMMA 3.1 (Lemma 3.1 from [16]). *There exists a constant $\zeta_* > 0$, independent of k, such that for any trial step $s_k^i \neq 0$ produced by a generalized pattern search method (Figure 3.2), we have $\| s_k^i \| \geq \zeta_* \Delta_k$.*

The next result is key to the convergence of pattern search methods. It states that the iterates produced by a pattern search method have a rigid algebraic structure.

THEOREM 3.2 (Theorem 3.2 from [16]). *Any iterate x_N produced by a generalized pattern search method (Figure 3.2) can be expressed in the following form:*

$$x_N = x_0 + \left(\beta^{r_{LB}} \alpha^{-r_{UB}}\right) \Delta_0 B \sum_{k=0}^{N-1} z_k, \tag{3.3}$$

where

- *x_0 is the initial guess;*
- *$\beta/\alpha \equiv \tau$, with $\alpha, \beta \in \mathbf{N}$ and relatively prime, and τ is as defined in the rules for updating Δ_k given in Figure 3.3;*
- *r_{LB} and r_{UB} are integers depending on N;*
- *Δ_0 is the initial choice for the step length control parameter;*

- *B is the basis matrix; and*
- *$z_k \in \mathbf{Z}^n$, $k = 0, \ldots, N-1$.*

The last result we recollect says, in conjunction with Lemma 3.1, that if we bound the size of the elements of the generating matrix (which is a reasonable thing to do), then Δ_k completely regulates the size of the steps a pattern search method takes. This result is a direct consequence of the fact that $s_k^i = \Delta_k B c_k^i$.

LEMMA 3.3 (Lemma 3.6 from [16]). *If there exists a constant $\mathcal{C} > 0$ such that for all k, $\mathcal{C} > \|c_k^i\|$, for all $i = 1, \ldots, p$, then there exists a constant $\psi_* > 0$, independent of k, such that for any trial step s_k^i produced by a generalized pattern search method (Figure 3.2) we have $\Delta_k \geq \psi_* \|s_k^i\|$.*

4. Convergence theory. We now present the first-order constrained stationary point convergence theory for pattern search methods for bound constrained problems. We begin by defining, for feasible x, the quantity

$$q(x) = P(x - g(x)) - x.$$

In the bound constrained theory the quantity $q(x)$ plays the role of $g(x)$ in the unconstrained theory, giving us a continuous measure of how close we are to constrained stationarity, as in the theory for methods based explicitly on derivatives (e.g., [5, 6, 8]). The following proposition summarizes two results concerning q that we will shortly need, particularly the fact that x is a constrained stationary point for (1.1) if and only if $q(x) = 0$. While stated for the particular domain Ω, the proposition holds for any closed convex domain. The results are classical; see section 2 of [8], for instance.

PROPOSITION 4.1. *Let $x \in \Omega$. Then*

$$\| \, q(x) \, \| \leq \| \, g(x) \, \|,$$

and x is a stationary point for problem (1.1) if and only if $q(x) = 0$.

We can now state the first convergence result for the general pattern search method for bound constrained minimization. Henceforth we will assume that $L_\Omega(x_0)$ is compact and that f is continuously differentiable on an open neighborhood D of $L_\Omega(x_0)$.

THEOREM 4.2. *Let $L_\Omega(x_0)$ be compact and suppose f is continuously differentiable on an open neighborhood D of $L_\Omega(x_0)$. Let $\{x_k\}$ be the sequence of iterates produced by a generalized pattern search method for bound constrained minimization (Figure 3.2). Then*

$$\liminf_{k \to +\infty} \| \, q(x_k) \, \| = 0\,.$$

The proof of this theorem is given in section 5.1, after we have established the necessary intermediate results.

We can strengthen the result given in Theorem 4.2 in the same way that we do in the unconstrained case [16]. First, we require the columns of the generating matrix C_k to remain bounded in norm, i.e., that there exists a constant $\mathcal{C} > 0$ such that for all k, $\mathcal{C} > \|c_k^i\|$, for all $i = 1, \ldots, p$. Second, we replace the original hypotheses on the result of the bound constrained exploratory moves with a stronger version, given in Figure 4.1. Third, we require that $\lim_{k \to +\infty} \Delta_k = 0$. All the algorithms described in section 6, except multidirectional search, satisfy this third condition because of the customary choice of $\Lambda = \{1\} \equiv \{\tau^0\}$. However, it is not necessary to force the steps to be nonincreasing.

1. $s_k \in \Delta_k P_k \equiv \Delta_k BC_k \equiv \Delta_k [B\Gamma_k \;\; BL_k]$.
2. $(x_k + s_k) \in \Omega$.
3. If $\min \{ f(x_k + y) \mid y \in \Delta_k B\Gamma_k,\; x_k + y \in \Omega \} < f(x_k)$, then $f(x_k + s_k) \le \min \{ f(x_k + y) \mid y \in \Delta_k B\Gamma_k,\; x_k + y \in \Omega \}$.

FIG. 4.1. *Strong hypotheses on the result of the bound constrained exploratory moves.*

THEOREM 4.3. *Let $L_\Omega(x_0)$ be compact and suppose f is continuously differentiable on an open neighborhood D of $L_\Omega(x_0)$. In addition, assume that the columns of the generating matrices are uniformly bounded in norm, that $\lim_{k\to+\infty} \Delta_k = 0$, and that the generalized pattern search method for bound constrained minimization (Figure 3.2) enforces the strong hypotheses on the result of the bound constrained exploratory moves (Figure 4.1). Then, for the sequence of iterates $\{x_k\}$ produced by the generalized pattern search method for bound constrained minimization,*

$$\lim_{k\to+\infty} \|q(x_k)\| = 0 \,.$$

The proof will be found in section 5.2.

5. Proof of Theorems 4.2 and 4.3. Throughout this section, x_k will refer to an iterate produced by a pattern search algorithm for bound constrained minimization. By design, x_k is feasible, i.e., $x_k \in \Omega$. Given an iterate x_k, let $g_k = g(x_k)$ and $q_k = q(x_k)$. Let $B(x,\delta)$ be the ball with center x and radius δ, and let ω denote the following modulus of continuity of $g(x)$: given $x \in L_\Omega(x_0)$ and $\varepsilon > 0$,

$$\omega(x,\varepsilon) = \sup \{ \delta > 0 \mid B(x,\delta) \subset D \text{ and } \| g(y) - g(x) \| < \varepsilon \text{ for all } y \in B(x,\delta) \} \,.$$

We begin with an elementary proposition concerning descent directions.

PROPOSITION 5.1. *Let $s \in \mathbf{R}^n$ and $x \in L_\Omega(x_0)$. Assume, too, that $g(x) \ne 0$ and $g(x)^T s \le -\varepsilon \| s \|$. Then, if $\| s \| < \omega(x, \frac{\varepsilon}{2})$,*

$$f(x+s) - f(x) \le -\frac{\varepsilon}{2} \| s \|.$$

Proof. If $\| s \| < \omega(x, \frac{\varepsilon}{2})$, then the closed line segment $[x, x+s]$ from x to $x+s$ is contained in D, where f is continuously differentiable. We may thus apply the mean-value theorem; we have, for some y on the line segment between x and $x+s$,

$$\begin{aligned} f(x+s) - f(x) &= g(x)^T s + (g(y) - g(x))^T s \\ &\le -\varepsilon \| s \| + \| g(y) - g(x) \| \| s \|. \end{aligned}$$

If $\| s \| < \omega(x, \frac{\varepsilon}{2})$, then $\| g(y) - g(x) \| \le \frac{\varepsilon}{2}$ and the result follows. ☐

It is in the proof of the next result that the bound constrained and the unconstrained cases differ most. The proof of Proposition 5.2 implicitly relies on the fact that in the bound constrained case, the directions in the pattern defined by the columns of BM_k are coordinate directions and thus are oriented normal and tangent to the faces of the feasible region. That this is not merely convenient is clear from the example given in section 2.

PROPOSITION 5.2. *Suppose that $q_k \ne 0$. Then there exists a $\nu_k > 0$ such that if $\Delta_k < \nu_k$, then there is a trial step s_k^i defined by a column of $\Delta_k B\Gamma_k$ for which $(x_k + s_k^i) \in \Omega$ and*

$$g_k^T s_k^i \le -n^{-\frac{1}{2}} \| q_k \| \| s_k^i \|.$$

Proof. We restrict our attention to the steps defined by the columns of $\Delta_k B\Gamma_k$; by hypothesis, $\Delta_k B\Gamma_k \equiv \Delta_k B[M_k \ -M_k] = \Delta_k[\mathrm{diag}(d_k^i) \ -\mathrm{diag}(d_k^i)]$ (see (3.2)). Choose an index m for which

$$| \, q_{k,m} \, | = \| \, q_k \, \|_\infty \geq n^{-\frac{1}{2}} \| \, q_k \, \|, \tag{5.1}$$

where $q_{k,m}$ is the mth component of q_k. Note that it is also the case that

$$| \, g_{k,m} \, | \geq | \, q_{k,m} \, | \tag{5.2}$$

and $\mathrm{sign}(g_{k,m}) = \mathrm{sign}(q_{k,m})$.

Let $s_k^i = -\mathrm{sign}(g_{k,m})\Delta_k \, | \, d_k^m \, | \, e_m$; this vector will be among the columns of $\Delta_k B\Gamma_k$. Since $x_k + q_k = P(x_k - g_k)$ is feasible, we have $\ell \leq x_k + q_k \leq u$ and thus

$$\ell_m \leq x_{k,m} + q_{k,m} \leq u_m.$$

It follows that if $\Delta_k \, | \, d_k^m \, | \leq | \, q_{k,m} \, |$, then the trial point $x_k^i = x_k + s_k^i$ will be feasible. Moreover, from (5.1) and (5.2),

$$g_k^T s_k^i = -\mathrm{sign}(g_{k,m})\Delta_k \, | \, d_k^m \, | \, g_{k,m} = -\| \, s_k^i \, \| \, | \, g_{k,m} \, | \ \leq -n^{-\frac{1}{2}} \| \, s_k^i \, \| \| \, q_k \, \|.$$

Defining $\nu_k = \| \, q_k \, \|_\infty / \, | \, d_k^m \, |$ then does the trick. $\square$

PROPOSITION 5.3. *Given any $\eta > 0$, there exists $\delta > 0$, independent of k, such that if $\Delta_k < \delta$ and $\| \, q_k \, \| > \eta$, the pattern search method for bound constrained minimization (Figure 3.2) will find an acceptable step s_k, i.e., $f(x_k + s_k) < f(x_k)$ and $(x_k + s_k) \in \Omega$.*

If, in addition, the columns of the generating matrix remain bounded in norm and we enforce the strong hypotheses on the result of the bound constrained exploratory moves (Figure 4.1), then, given any $\eta > 0$, there exist $\delta > 0$ and $\sigma > 0$, independent of k, such that if $\Delta_k < \delta$ and $\| \, q_k \, \| > \eta$, then

$$f(x_{k+1}) \leq f(x_k) - \sigma \| \, q_k \, \| \| \, s_k \, \|.$$

Proof. Since $g(x)$ is uniformly continuous on $L_\Omega(x_0)$ and $L_\Omega(x_0)$ is a compact subset of the open set D, there exists $\omega_* > 0$ such that

$$\omega\left(x_k, n^{-\frac{1}{2}}\eta\right) \geq \omega_*$$

for all k for which $\| \, q_k \, \| > \eta$.

Next, choose $d^* > 0$ such that $d_k^i \leq d^*$ for all i and k. This we can do because the set $\{d_k^i\}$ is finite (see (3.2) and the conditions on M_k given in section 3.1). Let

$$\nu_* = \frac{n^{-\frac{1}{2}}\eta}{d^*};$$

then

$$\nu_* = \frac{n^{-\frac{1}{2}}\eta}{d^*} \leq \frac{n^{-\frac{1}{2}}\| \, q_k \, \|}{d^*} \leq \frac{\| \, q_k \, \|_\infty}{d^*} \leq \nu_k$$

for all k for which $\| \, q_k \, \| > \eta$, where ν_k is as in Proposition 5.2.

Finally, let

$$\delta = \min\left(\nu_*, \omega_*/d^*\right).$$

Now suppose $\| q_k \| > \eta$ and $\Delta_k < \delta$. Since $\Delta_k < \nu_k$, Proposition 5.2 assures us of the existence of a step s_k^i defined by a column of $\Delta_k B\Gamma_k$ such that $(x_k + s_k^i) \in \Omega$ and

$$g_k^T s_k^i \le -n^{-\frac{1}{2}} \| q_k \| \| s_k^i \|.$$

At the same time, we also have

$$\| s_k^i \| \le \Delta_k d^* \le \omega_* \le \omega\left(x_k, n^{-\frac{1}{2}} \| q_k \|\right).$$

So, by Proposition 5.1,

$$f(x_k + s_k^i) - f(x_k) \le -\frac{1}{2} n^{-\frac{1}{2}} \| q_k \| \| s_k^i \|.$$

Thus, when $\Delta_k < \delta$, $f(x_k^i) \equiv f(x_k + s_k^i) < f(x_k)$ for at least one feasible $s_k^i \in \Delta_k B\Gamma_k$. The hypotheses on the result of the bound constrained exploratory moves guarantee that if

$$\min\{\ f(x_k + y)\ \mid\ y \in \Delta_k B\Gamma_k,\ x_k + y \in \Omega\ \} < f(x_k),$$

then $f(x_k + s_k) < f(x_k)$ and $(x_k + s_k) \in \Omega$. This proves the first part of the proposition.

If, in addition, we enforce the strong hypotheses on the result of the bound constrained exploratory moves, then we actually have

$$f(x_{k+1}) - f(x_k) \le -\frac{1}{2} n^{-\frac{1}{2}} \| q_k \| \| s_k^i \|.$$

Lemma 3.1 then ensures that

$$f(x_{k+1}) \le f(x_k) - \frac{1}{2} n^{-\frac{1}{2}} \zeta_* \Delta_k \| q_k \|.$$

Applying Lemma 3.3, we arrive at

$$f(x_{k+1}) \le f(x_k) - \sigma \| q_k \| \| s_k \|,$$

where $\sigma = \frac{1}{2} n^{-\frac{1}{2}} \zeta_* \psi_*$. ∎

COROLLARY 5.4. *If* $\liminf_{k \to +\infty} \| q_k \| \neq 0$*, then there exists a constant* $\Delta_* > 0$ *such that for all* k*,* $\Delta_k > \Delta_*$*.*

Proof. By hypothesis, there exists K and $\eta > 0$ such that for all $k > K$, $\| q_k \| > \eta$. By Proposition 5.3, we can find δ such that if $k > K$ and $\Delta_k < \delta$, then we will find an acceptable step. In view of the rules for updating Δ_k given in Figure 3.3, we are assured that for all $k > K$, $\Delta_k > \theta\delta$. We may then take $\Delta_* = \min\{\Delta_0, \ldots, \Delta_K, \theta\delta\}$. ∎

The next theorem combines the strict algebraic structure of the iterates with the simple decrease condition of the generalized pattern search algorithm for bound constrained problems (Figure 3.2), along with the rules for updating Δ_k (Figure 3.3), to give us a useful fact about the limiting behavior of Δ_k.

THEOREM 5.5. *Assume that* $L_\Omega(x_0)$ *is compact. Then* $\liminf_{k \to +\infty} \Delta_k = 0$.

Proof. The proof is like that of Theorem 3.3 in [16]. Suppose $0 < \Delta_{LB} \leq \Delta_k$ for all k. Using the rules for updating Δ_k found in Figure 3.3, it is possible to write Δ_k as $\Delta_k = \tau^{r_k}\Delta_0$, where $r_k \in \mathbf{Z}$.

The hypothesis that $\Delta_{LB} \leq \Delta_k$ for all k means that the sequence $\{\tau^{r_k}\}$ is bounded away from zero. Meanwhile, we also know that the sequence $\{\Delta_k\}$ is bounded above because all the iterates x_k must lie inside the set $L_\Omega(x_0) = \{x \in \Omega : f(x) \leq f(x_0)\}$ and the latter set is compact; Lemma 3.1 then guarantees an upper bound Δ_{UB} for $\{\Delta_k\}$. This, in turn, means that the sequence $\{\tau^{r_k}\}$ is bounded above. Consequently, the sequence $\{\tau^{r_k}\}$ is a finite set. Equivalently, the sequence $\{r_k\}$ is bounded above and below.

Next we recall the exact identity of the quantities r_{LB} and r_{UB} in Theorem 3.2; the details are found in the proof of Theorem 3.3 in [16]. At iteration N we have

$$r_{LB} = \min_{0 \leq k < N}\{r_k\}, \qquad r_{UB} = \max_{0 \leq k < N}\{r_k\}.$$

If, in the matter at hand, we let

$$r_{LB} = \min_{0 \leq k < +\infty}\{r_k\}, \qquad r_{UB} = \max_{0 \leq k < +\infty}\{r_k\}, \tag{5.3}$$

then (3.3) holds for the bounds given in (5.3), and we see that for all k, x_k lies in the translated integer lattice G generated by x_0 and the columns of $\beta^{r_{LB}}\alpha^{-r_{UB}}\Delta_0 B$.

The intersection of the compact set $L_\Omega(x_0)$ with the lattice G is finite. Thus, there must exist at least one point x_* in the lattice for which $x_k = x_*$ for infinitely many k.

We now appeal to the simple decrease condition (c) in Figure 3.2, which guarantees that an iterate cannot be revisited infinitely many times since we accept a new step s_k if and only if $f(x_k) > f(x_k + s_k)$ and $(x_k + s_k) \in \Omega$. Thus there exists an N such that for all $k \geq N$, $x_k = x_*$, which implies that $f(x_k) = f(x_k + s_k)$.

We now appeal to the rules for updating Δ_k (Figure 3.3, part (a)) to see that $\Delta_k \to 0$, thus leading to a contradiction. ∎

5.1. Proof of Theorem 4.2. The proof is like that of Theorem 3.5 in [16]. Suppose that $\liminf_{k\to+\infty} \| q(x_k) \| \neq 0$. Then Corollary 5.4 tells us that there exists $\Delta_* > 0$ such that for all k, $\Delta_k \geq \Delta_*$. But this contradicts Theorem 5.5.

5.2. Proof of Theorem 4.3. The proof, also by contradiction, follows that of Theorem 3.7 in [16]. Suppose $\limsup_{k\to+\infty} \| q(x_k) \| \neq 0$. Let $\varepsilon > 0$ be such that there exists a subsequence $\| q(x_{m_i}) \| \geq \varepsilon$. Since

$$\liminf_{k\to+\infty} \| q(x_k) \| = 0,$$

given any $0 < \eta < \varepsilon$, there exists an associated subsequence l_i such that

$$\| q(x_k) \| \quad > \eta \qquad \text{for} \qquad m_i \leq k < l_i, \qquad \| q(x_{l_i}) \| \quad < \quad \eta.$$

Since $\Delta_k \to 0$, we can appeal to Proposition 5.3 to obtain for $m_i \leq k < l_i$, i sufficiently large,

$$f(x_k) - f(x_{k+1}) \quad \geq \quad \sigma\| q(x_k) \|\| s_k \| \quad \geq \quad \sigma\eta\| s_k \|,$$

where $\sigma > 0$. Then the telescoping sum,

$$\begin{aligned}(f(x_{m_i}) - f(x_{m_i+1})) + (f(x_{m_i+1}) - f(x_{m_i+2})) + \cdots + (f(x_{l_i-1}) - f(x_{l_i}))\\ \geq \sum_{k=m_i}^{l_i} \sigma\eta\| s_k \|,\end{aligned}$$

gives us

$$f(x_{m_i}) - f(x_{l_i}) \quad \geq \quad \sum_{k=m_i}^{l_i} \sigma\eta \| \, s_k \, \| \quad \geq \quad c' \| \, x_{m_i} - x_{l_i} \, \|.$$

Since f is bounded below, $f(x_{m_i}) - f(x_{l_i}) \to 0$ as $i \to +\infty$, so $\| \, x_{m_i} - x_{l_i} \, \| \to 0$ as $i \to +\infty$. Then, because q is uniformly continuous, $\| \, q(x_{m_i}) - q(x_{l_i}) \, \| < \eta$ for i sufficiently large. However,

$$\| \, q(x_{m_i}) \, \| \quad \leq \quad \| \, q(x_{m_i}) - q(x_{l_i}) \, \| + \| \, q(x_{l_i}) \, \| \quad \leq \quad 2\eta. \tag{5.4}$$

Since (5.4) must hold for any η, $0 < \eta < \varepsilon$, we have a contradiction (e.g., try $\eta = \frac{\varepsilon}{4}$).

6. Examples of pattern search methods for bound constrained minimization. A section of [16] is devoted to showing that each of the following four algorithms are pattern search methods for unconstrained minimization:

- coordinate search with fixed step lengths,
- evolutionary operation using two-level factorial designs [1, 3, 14],
- the original pattern search method of Hooke and Jeeves [9], and
- the multidirectional search algorithm of Dennis and Torczon [7, 15].

In this section we will discuss how these algorithms may be extended to bound constrained problems. We shall see that coordinate search and the pattern search method of Hooke and Jeeves extend without modification to the bound constrained case. Conversely, in the case of multidirectional search, we must require the initial basis matrix to be a diagonal matrix (in the unconstrained case, we can allow any nonsingular basis matrix); in addition, we must augment the columns of the generating matrix to ensure a sufficient set of search directions. In the case of evolutionary operation, we also must augment the columns of the generating matrix, which we do using a classical variant of factorial designs [2].

The difference between pattern search methods for unconstrained problems and bound constrained problems lies in the two additional conditions discussed in section 3.5. First, pattern search methods for bound constrained problems must start with a feasible iterate and choose feasible trial steps. Second, the core pattern BM_k must be defined by a diagonal matrix.

We assume that we begin with a feasible iterate; by design, pattern search methods for bound constrained problems thereafter accept only feasible iterates. Thus, the only thing we need to check is that the core pattern BM_k is defined by a diagonal matrix.

It is this latter condition that causes us to restrict the admissible choice of the basis matrix in multidirectional search and then augment the columns of the generating matrix. Moreover, G.E.P. Box's method of evolutionary operation using two-level factorial designs does not satisfy this diagonality condition; in section 2 we presented a simple counterexample that showed how evolutionary operation can fail as a consequence in the bound constrained case.

6.1. Coordinate search and the pattern search method of Hooke and Jeeves. Coordinate search and the pattern search method of Hooke and Jeeves extend to bound constrained problems without change. In both cases the basis matrix B is typically chosen to be a diagonal matrix: either the identity or a matrix whose entries reflect the relative scaling of the variables. Furthermore, the first 3^n columns of C_k, which are fixed for all iterations k of both algorithms, are composed of all possible combinations of $\{-1, 0, 1\}$. In [16] these columns are organized so that the first $2n$ consist of the identity matrix I and its negative $-I$. In terms of our formalism, then, $M_k = I$ for all iterations k. It follows that BM_k is a diagonal matrix, as required.

6.2. Evolutionary operation using factorial design. In section 2 a simple example sufficed to show that evolutionary operation cannot be used for bound constrained minimization without alteration. In terms of our formalism, the problem is the following: For the evolutionary operation algorithm using factorial designs, the basis matrix B is usually selected to be the identity or a diagonal matrix chosen so that the entries along the diagonal represent the relative scaling among the variables. However, this convention is not sufficient to ensure that BM_k is a diagonal matrix. The problem lies with the generating matrix $C = [M \ -M \ L]$. (The generating matrix C is fixed across all iterations of evolutionary operation.) The generating matrix contains in its columns all possible combinations of $\{-1, 1\}$ to which is appended a column of zeros. Clearly, no subset of n columns of C can be chosen to form a diagonal matrix M.

As noted in section 2, one remedy would be to use a composite design [2]. An example of such a design that satisfies the requirements of the bound constrained global convergence theory would be to choose M to be the diagonal matrix with entries of 2 along the diagonal. These $2n$ columns augment the original pattern of factorial design. This was illustrated in Figure 2.3.

6.3. Multidirectional search. The reader should be forewarned that our description and discussion of multidirectional search take a point of view that is ostensibly at odds with the formalism of section 3.1. The generating matrix Γ is viewed as fixed; typically $\Gamma = [M \ -M] \equiv [I \ -I]$. The basis matrix, on the other hand, is viewed as varying from iteration to iteration so that B_k corresponds to the edges in the current simplex that are adjacent to the current iterate x_k. This is the reverse of the discussion in section 3.1, where B is fixed and Γ_k varies. However, the former view of multidirectional search is not incompatible with the formalism of pattern search methods, as noted in [16], and as we shall have reason to discuss here.

The extension of multidirectional search to problems with bound constraints requires us to restrict the choice of a starting simplex and to augment the columns of the generating matrix.

The first restriction is minor and is usually satisfied by the customary choices made in practice. In multidirectional search, the columns of B_0 are formed from the edges of an initial simplex adjacent to the initial iterate x_0, which is one of the $n+1$ vertices of the simplex. In the case of bound constraints, we restrict the starting simplex to be a right-angled simplex; i.e., the vertices of the simplex are x_0 and the points $x_0 + \alpha_i e_i$, where $\alpha_i \in \mathbf{R}$ and $i = 1, \ldots, n$. Because of this choice, $B_0 = \mathrm{diag}(\alpha_i)$. Since $M \equiv I$, the product $B_0 M$ is a diagonal matrix.

However, even if the initial simplex is restricted to be a right-angled simplex so that $B_0 M$ is diagonal, there is no guarantee that in subsequent iterations $B_k M$ will be diagonal. To understand why this is so, and how this may be corrected by augmenting the columns of the generating matrix, we need to discuss how multidirectional search fits within the formalism of pattern search methods. These details are absent from [16], so we present them here.

At iteration k, the basis matrix is

$$B_k = \left[b_k^1 \cdots b_k^n\right] = \left[(v_k^1 - v_k^0) \cdots (v_k^n - v_k^0)\right],$$

where v_k^i, $i = 0, \ldots, n$, are the vertices of the simplex associated with multidirectional search at this iteration. Define

$$T_i = \begin{cases} I, & i = 0, \\ -\left(I - e_i e_i^T - \sum_{m=1}^n e_i e_m^T\right), & i = 1, \ldots, n. \end{cases}$$

Now consider what happens in the next iteration. If the iteration is unsuccessful, then $v_{k+1}^0 = v_k^0$ and the new basis for the pattern, which is determined by the edges of the simplex emanating from v_{k+1}^0, is

$$B_{k+1} = B_k = B_k T_0.$$

If, on the other hand, the iteration is successful, then $v_{k+1}^0 = v_k^0 - (v_k^j - v_k^0)$ for some $j \in \{1, \ldots, n\}$, and the new basis will be the set of vectors

$$b_{k+1}^i = \begin{cases} b_k^j & \text{if } i = j, \\ -b_k^i + b_k^j & \text{otherwise.} \end{cases}$$

In this case,

$$B_{k+1} = B_k T_j.$$

Thus, in general,

$$B_{k+1} = B_k T_{j_{k+1}}, \tag{6.1}$$

and so

$$B_k = B_{k-1} T_{j_k} = B_{k-2} T_{j_{k-1}} T_{j_k} = \cdots = B_0 \prod_{i=1}^k T_{j_i}. \tag{6.2}$$

Our next goal is to simplify this relation further.

First note that

$$T_\ell e_i = \begin{cases} e_\ell & \text{if } i = \ell, \\ e_\ell - e_i & \text{if } i \neq \ell. \end{cases} \tag{6.3}$$

Let $E(i, \ell)$ denote the elementary permutation matrix that swaps the ith and ℓth columns when acting on matrices from the right; we have

$$E(i, \ell) = I - e_i e_i^T - e_\ell e_\ell^T + e_\ell e_i^T + e_i e_\ell^T.$$

Using (6.3), we find that if $i \neq \ell$, then

$$T_\ell E(i, \ell) = T_\ell + e_i e_i^T - e_i e_\ell^T \tag{6.4}$$

and

$$\left(T_i \left(-T_\ell\right)\right) e_i = e_\ell. \tag{6.5}$$

Meanwhile, a short calculation shows that for $i, \ell = 1, \ldots, n$,

$$T_i T_\ell = I - e_\ell e_\ell^T - \sum_{m=1}^n e_\ell e_m^T - e_i e_i^T + \delta_\ell^i e_i e_\ell^T + \delta_\ell^i \sum_{m=1}^n e_i e_m^T + e_i e_\ell^T,$$

where δ_ℓ^i is the Kronecker delta. If $i = \ell$, this reduces to

$$T_i T_i = I, \tag{6.6}$$

and if $i \neq \ell$, using (6.4) we obtain

$$\begin{aligned} T_i T_\ell &= I - e_\ell e_\ell^T - \sum_{m=1}^{n} e_\ell e_m^T - e_i e_i^T + e_i e_\ell^T \\ &= -T_\ell - e_i e_i^T + e_i e_\ell^T = -T_\ell E(i,\ell). \end{aligned} \tag{6.7}$$

From (6.6) and (6.7) we obtain the rule

$$T_i T_\ell = \begin{cases} I & \text{if } i = \ell, \\ -T_\ell E(i,\ell) & \text{otherwise.} \end{cases} \tag{6.8}$$

We can then use (6.8) to reduce (6.2) to

$$B_k = \pm B_0 T_{\ell_k} \Pi_k$$

for some T_{ℓ_k} and permutation matrix Π_k.

This relationship reveals several things. The first is that it reconciles the usual description of multidirectional search with the formal abstract definition of a pattern search method; the pattern matrix is given by

$$B_k C = \pm B_0 T_{j_k} \Pi_k [I \ -I \ 0] = B_0 [T_{j_k} \ \ -T_{j_k} \ 0] \Pi_k \equiv B C_k. \tag{6.9}$$

That is, we may interpret multidirectional search in terms of a fixed basis B and a changing generating matrix C_k.

We can also see that while $B\Gamma_0$ will be diagonal, this diagonality may be lost in subsequent iterations. However, the form of the generic pattern from the unconstrained algorithm suggests one way to circumvent this problem in the bound constrained case. This remedy will, moreover, preserve the geometric interpretation of the pattern in multidirectional search in terms of a simplex.

First, if we ignore the permutation in (6.9), which affects only column ordering, the pattern at iteration k in the unconstrained case is given by

$$B_k C \equiv B C_k = B_0 [T_{j_k} \ \ -T_{j_k} \ 0].$$

Suppose we augment the columns of C to include all the T_i:

$$C \quad = \quad [-T_0 \ -T_1 \ \cdots \ -T_n \ 0].$$

At any iteration k, up to a column permutation, the basis matrix is the matrix $B_k = \pm B T_{j_k}$, $j_k \in \{0, \ldots, n\}$. When we then form the pattern $P_k = \Delta_k B_k C$, we have

$$P_k \quad = \quad \Delta_k B_k C \quad = \quad \Delta_k B [\pm T_{j_k} T_0 \ \pm T_{j_k} T_1 \ \cdots \ \pm T_{j_k} T_n \ 0] \quad \equiv \quad \Delta_k B C_k.$$

Now note that (6.5) means that for $j_k \neq l$, the j_kth column of $-T_{j_k} T_\ell$ is the ℓth basis vector. Consequently, we are guaranteed that by a permutation of the columns of C_k,

$$C_k = [I \ -I \ L_k] \equiv [\Gamma \ L_k],$$

where L_k changes at each iteration, but Γ does not. Since we require the initial simplex to be a right-angled simplex, we may then be assured that $B\Gamma = [\mathrm{diag}(\alpha_i)\ -\mathrm{diag}(\alpha_i)]$, as required.

Moreover, this augmentation of C and the search through its columns can be implemented in a way that preserves the relationship of the pattern to the moving simplex that characterizes multidirectional search. This is possible because the matrices T_i, $i = 0, \ldots, n$, capture how the basis changes in association with a change of simplex. This is the gist of (6.1). The implications for any implementation of this modification to multidirectional search to handle bound constraints will appear elsewhere.

7. Conclusion. We have presented a reasonable extension of pattern search methods for unconstrained minimization to bound constrained problems. The extension is supported by a global convergence theory as strong as that for the unconstrained case. The generalization imposes few additional requirements, and as we have seen in section 6, the classical pattern search methods for unconstrained minimization or straightforward variants thereof carry over to the bound constrained case.

One issue we have not discussed is that of identifying active constraints, as in [4, 5]. One would wish to show that if the sequence $\{x_k\}$ converges to a nondegenerate stationary point x_*, then in a finite number of iterations the iterates x_k land on the constraints active at x_* and remain thereafter on those constraints.

There are three difficulties in proving such a result for pattern search methods for bound constrained minimization. The first is minor. If the iterates x_k are to identify the active constraints for a stationary point on the boundary of the feasible region, we must ensure that the lattice manifest in Theorem 3.2 actually allows iterates to land on the boundary. This requires additional but straightforward conditions on such quantities as x_0, τ, Δ_0, and the pattern matrices P_k (see, for instance, [17]). A related but more subtle difficulty is that the relative sizes of the steps in the core pattern and the remaining points in the pattern must obey certain relations in order to ensure that the algorithm does not take a purely interior approach to a point on the boundary. This rules out, for instance, certain of the composite designs suggested by Box and Wilson [2].

The most serious obstacle is showing that ultimately the iterates will land on the active constraints and remain there. For algorithms such as those considered in [4, 5], this is not a problem because the explicit use of the gradient impels the iterates to do this in the neighborhood of a nondegenerate stationary point. However, pattern search methods do not have this information. On the other hand, the kinship of pattern search methods and gradient projection methods makes us hopeful that ultimately we will be able to prove that pattern search methods also identify the active constraints in a finite number of iterations.

One can also extend pattern search methods to linearly constrained minimization [11]. The specification of pattern search methods for handling general linear inequalities is more involved, and the analysis is lengthier and more complicated. For bound constrained problems the analysis is enormously simplified because of the straightforward geometry of the feasible region and the fact that we know the explicit form of the projected gradient.

Acknowledgments. We wish to thank Margaret Wright for her helpful comments, particularly her suggestion that we include an introductory example that provides some intuition about how these methods work. We also wish to thank Chen Xin for catching an error in an earlier version of the paper.

REFERENCES

[1] G. E. P. Box, *Evolutionary operation: A method for increasing industrial productivity*, Appl. Statist., 6 (1957), pp. 81–101.

[2] G. E. P. Box and K. B. Wilson, *On the experimental attainment of optimum conditions*, J. Roy. Statist. Soc. Ser. B, XIII (1951), pp. 1–45.

[3] M. J. Box, D. Davies, and W. H. Swann, *Non-Linear Optimization Techniques*, ICI Monograph 5, Oliver & Boyd, Edinburgh, Scotland, 1969.

[4] J. V. Burke and J. J. Moré, *On the identification of active constraints*, SIAM J. Numer. Anal., 25 (1988), pp. 1197–1211.

[5] P. H. Calamai and J. J. Moré, *Projected gradient methods for linearly constrained problems*, Math. Programming, 39 (1987), pp. 93–116.

[6] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 25 (1988), pp. 433–460.

[7] J. E. Dennis, Jr. and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448–474.

[8] J. C. Dunn, *Global and asymptotic convergence rate estimates for a class of projected gradient processes*, SIAM J. Control Optim., 19 (1981), pp. 368–400.

[9] R. Hooke and T. A. Jeeves, *Direct search solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–229.

[10] D. L. Keefer, *Simpat: Self-bounding direct search method for optimization*, Indust. Engrg. Chem. Process Design Develop., 12 (1973), pp. 92–99.

[11] R. M. Lewis and V. J. Torczon, *Pattern search methods for linearly constrained minimization*, SIAM J. Optim., to appear.

[12] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[13] W. Spendley, G. R. Hext, and F. R. Himsworth, *Sequential application of simplex designs in optimisation and evolutionary operation*, Technometrics, 4 (1962), pp. 441–461.

[14] W. H. Swann, *Direct search methods*, in Numerical Methods for Unconstrained Optimization, W. Murray, ed., Academic Press, London, New York, 1972, pp. 13–28.

[15] V. Torczon, *Multi-Directional Search: A Direct Search Algorithm for Parallel Machines*, Ph.D. thesis, Department of Mathematical Sciences, Rice University, Houston, TX, 1989; available as Tech. report 90-07, Department of Computational and Applied Mathematics, Rice University, Houston, TX.

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

[17] M. W. Trosset and V. Torczon, *Numerical Optimization Using Computer Experiments*, ICASE Report No. 97-38, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA, 1997.