

RANK ORDERING AND POSITIVE BASES IN PATTERN SEARCH ALGORITHMS *

ROBERT MICHAEL LEWIS † AND VIRGINIA TORCZON ‡

Abstract. We present two new classes of pattern search algorithms for unconstrained minimization: the rank ordered and the positive basis pattern search methods. These algorithms can nearly halve the worst case cost of an iteration compared to the classical pattern search algorithms. The rank ordered pattern search methods are based on a heuristic for approximating the direction of steepest descent, while the positive basis pattern search methods are motivated by a generalization of the geometry characteristic of the patterns of the classical methods. We describe the new classes of algorithms and present the attendant global convergence analysis.

Key words. direct search methods, pattern search, positive linear dependence

1. Introduction. In this paper we introduce two new classes of pattern search algorithms: the *rank ordered* and *positive basis* pattern search methods for the unconstrained minimization problem

$$\underset{x \in \mathbf{R}^n}{\text{minimize}} \quad f(x).$$

The rank ordered and positive basis pattern search methods extend the analysis developed in [12]. These new classes of algorithms can almost halve the worst case cost of an iteration when compared with the classical pattern search methods considered in [12]. Moreover, the simple heuristics that motivate rank ordered and positive basis pattern search methods are intuitively appealing and make the methods straightforward to describe.

Pattern search methods form a class of “steep descent” procedures (a term we will shortly explain) for nonlinear minimization. Examples include [1], [6], and [11]. While these algorithms have no explicit recourse to a Taylor series model of the objective, or any information about directional derivatives, one can develop a global convergence analysis for pattern search methods with results similar to those for quasi-Newton methods.

This is possible because pattern search methods are gradient-related, so that when the steps become small enough, they are guaranteed to capture a portion of the improvement promised by the steepest descent direction. This intuition seems to have been in the minds of the early developers of the broader class of direct search methods; for instance, Spendley, Hext, and Himsworth [9] refer to their algorithm as a method of “steep ascent” (they were considering maximization) to indicate its kinship to the method of steepest ascent. The convergence analysis for pattern search methods confirms that this is particularly apt and we therefore propose the term “methods of steep descent” to describe pattern search methods.

*Revision: 1.8 Date: 1998/01/13 22:55:34

†This research was supported by the National Aeronautics and Space Administration under NASA Contract No. 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 23681-0001, buckaroo@icase.edu.

‡Department of Computer Science, College of William and Mary, Williamsburg, Virginia 23187-8795, va@cs.wm.edu. This research was supported by the Air Force Office of Scientific Research grant F49620-95-1-0210 and by the National Aeronautics and Space Administration under NASA Contract No. 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 23681-0001.

A characterization and analysis of a class of pattern search methods is presented in [12]. Key to the global convergence analysis is the fact that these methods produce iterates that lie on a rational lattice. To ensure global convergence, the pattern search methods discussed in [12] must satisfy a condition that prevents the mesh size from being reduced if one of the steps in a set of $2n$ *core* steps produces decrease in the objective value. This condition allows the acceptance of *any* step that lies in a certain finite subset of the lattice as long as the step produces simple decrease on the objective value at the current iterate, and plays the role in the global convergence analysis of pattern search methods that is played by the fraction of Cauchy decrease condition for trust-regions and the Armijo-Goldstein-Wolfe conditions for line-searches.

As a consequence, one can construct pattern search algorithms that require, in the best case, only *one* new objective value per iteration. This feature is used to advantage in a proposed strategy for scientific and engineering optimization [4] for problems in which the computational cost of a single objective evaluation is sufficiently great as to merit particular care in choosing steps at which to evaluate the objective. On the other hand, the worst case occurs when it is necessary to reduce the mesh size to make further progress, for instance, because the search has reached a neighborhood either that contains a local minimizer or where the objective is highly nonlinear. For the classical pattern search algorithms studied in [12], as many as $2n$ objective evaluations may be necessary in this case.

The rank ordered pattern search methods and positive basis pattern search methods that we introduce here improve this worst case bound to $n + 1$ objective values. They do this by reducing to a minimal size the set of directions that we must consider to be assured of having a sufficiently rich set of search directions to capture a suitably large component of the direction of steepest descent.

The *rank ordered* pattern search methods are motivated by the following heuristic. Suppose that in addition to the objective value at the current iterate x_k , we know the objective value at n other suitably independent points. Then the direction from the point with the highest objective value to the point with the lowest objective value should be, when this pattern of points is sufficiently small, a crude estimate of the direction of steepest descent. Unlike a finite-difference approximation to the direction of steepest descent, this approximation ignores the distances between the points (and thus the relative rate of change). On the other hand, this estimate only requires that we identify the best and worst objective values among a set of $n + 1$ values. This is in keeping with a distinctive feature of pattern search methods: namely, that they only require information about the relative rank of objective values, and can actually be used in the absence of any numerical objective value, as might be the case when the algorithm is driven by a subjective preference on the part of the user.

The analysis in [12] does not make use of rank-order information, though it is implicit in the multidirectional search algorithm considered there. As it happens, the simple heuristic of using rank ordering and the notion of “steep descent”, described above, to suggest search directions suffices to prove global convergence. We note that this heuristic also motivates the direct search method of Nelder and Mead, which is not a pattern search method and is known not to be robust [8], and the aforementioned direct search method of Spendley, Hext, and Himsworth [9], the very interesting analysis of which we will discuss elsewhere.

The heuristic of using the best and worst objective values to suggest a direction of steep descent reduces the cardinality of the core set of steps from $2n$ to $n + 1$, but may introduce an inherently sequential component into the algorithm. It may

be the case that we cannot specify the final search direction until after the first n points have been evaluated and ranked. The *positive basis* pattern search methods have the attractive feature that they avoid this sequentiality and thus are well-suited for parallel implementation.

It was in the course of re-examining the work on direct search methods by Yu Wen-ci [13, 14] that we realized the utility of the theory of positive linear dependence developed by C. Davis in [3] for generalizing pattern search methods in a useful way. One can view the positive basis pattern search methods as the natural generalization of the algorithms considered in [12].

We say “natural” for the following reason. In the class of algorithms studied in [12], the $2n$ core directions played the technical role of ensuring that we would search in a direction that made a positive inner product with the direction of steepest descent. The notion of a *positive basis* [3] is the correct way to generalize this latter property, and allows us to reduce, *a priori*, the cardinality of the core set of steps from $2n$ to as few as $n + 1$. Because this core set can be specified in advance, we avoid the sequential element of the rank ordered pattern search methods.

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

All norms are Euclidean vector norms or the associated operator norm. We define $L(x) = \{y : f(y) \leq f(x)\}$. We will denote the gradient $\nabla f(x)$ by $g(x)$ and the gradient at iteration k $g(x_k)$ by g_k .

By abuse of notation, if A is a matrix, $y \in A$ means that the vector y is a column of A . We will also use c and C to denote divers constants whose identity will vary from place to place but whose nature will never depend on the iteration k .

2. Pattern Search Methods. Pattern search methods, including the two new classes of algorithms under discussion, are characterized by the nature of the generating matrices and the exploratory moves algorithms. These features are discussed more fully in [12].

To define a pattern we need two components, a *basis matrix* and a *generating matrix*. We will expand the class of pattern search algorithms by expanding the class of admissible generating matrices.

The basis matrix can be any nonsingular matrix $B \in \mathbf{R}^{n \times n}$. The generating matrix is a matrix $C_k \in \mathbf{Z}^{n \times p_k}$, where $p_k > n + 1$. We partition the generating matrix into components

$$(1) \quad C_k = \begin{bmatrix} \Gamma_k & L_k & 0 \end{bmatrix}.$$

We require that $\Gamma_k \in \mathbf{M}$, where \mathbf{M} is a finite set of integral matrices with full row rank. In §2.1 and §2.2 we will discuss further requirements on the members of \mathbf{M} ; we will see that Γ_k must have at least $n + 1$ columns. The 0 in the last column of C_k is a single 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 (1) to partition P_k as follows:

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

Given $\Delta_k \in \mathbf{R}$, $\Delta_k > 0$, we define a *trial step* s_k^i to be any vector of the form $s_k^i = \Delta_k Bc_k^i$, where c_k^i is a column of C_k . Note that Bc_k^i determines the direction of the step, while Δ_k serves as a step length parameter.

At iteration k , we define a *trial point* as any point of the form $x_k^i = x_k + s_k^i$, where x_k is the current iterate.

Algorithm 1 states the pattern search method for unconstrained minimization. To define a particular pattern search method, we must specify a basis matrix B , the generating matrices C_k , the exploratory moves to be used to produce a step s_k , and the algorithms for updating C_k and Δ_k .

Algorithm 1. The pattern search method for unconstrained minimization.

Let $x_0 \in \mathbf{R}^n$ and $\Delta_0 > 0$ be given.

For $k = 0, 1, \dots$,

- a) Compute $f(x_k)$.
- b) Determine a step s_k using an unconstrained 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 .

If $f(x_k + s_k) < f(x_k)$ we call the iteration *successful*; otherwise, we call the iteration *unsuccessful*.

We have the following Hypotheses on Unconstrained Exploratory Moves. This is the same as in [12]. The specification of the matrix Γ_k in (1) is different from that in [12], however. There, Γ_k was required to be a $n \times 2n$ matrix with full row rank. The Hypotheses on Unconstrained Exploratory Moves require that before an iteration is declared unsuccessful, we must examine all the steps in the *core pattern* determined by $\Delta_k B \Gamma_k$ for a lower objective value. Thus, for the algorithms considered in [12], in the worst case one would need to compute a minimum of $2n$ objective values in an iteration. In the rank ordered and positive basis pattern search methods, however, the size of Γ_k can be as small as $n \times (n + 1)$, which has the effect of reducing the worst case cost of an iteration to $n + 1$ objective evaluations.

Hypotheses on Unconstrained Exploratory Moves.

1. $s_k \in \Delta_k P_k \equiv \Delta_k B C_k \equiv \Delta_k [B \Gamma_k \ B L_k \ 0]$.
2. If $\min \{ f(x_k + y) \mid y \in \Delta_k B \Gamma_k \} < f(x_k)$, then $f(x_k + s_k) < f(x_k)$.

Algorithm 2 specifies the rule for updating Δ_k . 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.

Algorithm 2. Updating Δ_k .

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

- 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 $\Delta_{k+1} = \lambda_k \Delta_k$.

2.1. Rank Ordered Pattern Search Methods. As discussed in the Introduction, one can use the best and worst objective values in a portion of the pattern to suggest an intuitive direction of steepest descent for exploration. This heuristic, which we will develop formally here, ensures that when Δ_k is sufficiently small, one has a trial step that is a suitably good direction of descent.

For the rank ordered pattern search methods, the generating matrix is an $n \times p_k$

matrix, $p_k > n + 1$, which we partition as:

$$C_k = [S_k \ R_k \ L_k \ 0] = [\Gamma_k \ L_k \ 0].$$

We require S_k to be a nonsingular element of $\mathbf{Z}^{n \times n}$. The notation S_k is chosen to suggest *simplex*, and R_k is meant to suggest *reflection*, as in the multidirectional search algorithm [11].

Let $S_k = [d_k^1 \ d_k^2 \ \cdots \ d_k^n]$ and consider the simplex with vertices

$$(2) \quad \{x_k, x_k + \Delta_k B d_k^1, x_k + \Delta_k B d_k^2, \dots, x_k + \Delta_k B d_k^n\} = \{v_k^0, v_k^1, v_k^2, \dots, v_k^n\},$$

where the vertices are ordered (and possibly relabeled) so that

$$(3) \quad f(v_k^0) \leq f(v_k^1), \dots, f(v_k^{n-1}) \leq f(v_k^n).$$

Note that we only need to identify the vertices with the best and the worst objective values; we are not required to give a relative ranking of the remaining $n - 1$ vertices.

Given the best and worst objective values among the vertices of the simplex (2), we can then say how R_k is chosen. We require R_k to contain a column of the form

$$(4) \quad \sum_{i=1}^n \rho_k^i \Delta_k^{-1} B^{-1} (v_k^0 - v_k^i), \quad \rho_k^i \geq 0, \rho_k^n > 0.$$

The presence of a column of this form in R_k means that the core pattern $\Delta_k B \Gamma_k$ will contain a trial step of the form

$$(5) \quad \sigma_k = \sum_{i=1}^n \rho_k^i (v_k^0 - v_k^i).$$

Since the column defined by (4) is contained in $\Gamma_k = [S_k \ R_k]$, there are, implicitly, further restrictions on the choice of ρ_k^i , $i = 1, \dots, n$, because $\Gamma_k \in \mathbf{M}$, and \mathbf{M} is a finite set of integral matrices with full row rank. One choice would be to set $\rho_k^i = 0$ for $i = 1, \dots, n - 1$ and $\rho_k^n = 1$. An example can be seen for \mathbf{R}^2 in Fig. 4. Another choice would be to set $\rho_k^i = 1/n$ for $i = 1, \dots, n$, provided that the R_k that results is integral. This can be arranged if one is willing to scale B appropriately. The latter choice gives equal weight to all the n potential descent directions $(v_k^i - v_k^0)$ for $i = 1, \dots, n$.

The requirement that $\rho_k^n > 0$ ensures that at least one of the search directions defined by $B R_k$ is biased towards the direction from the worst vertex to the best. This distinguished direction will, when Δ_k is small, capture enough of the direction of steepest descent to allow us to prove global convergence.

Here we see how using the rank-order information and the steepest edge heuristic can introduce a sequential element into the definition of C_k . For instance, the choice $\rho_k^i = 0$ for $i = 1, \dots, n - 1$ and $\rho_k^n = 1$ means that we cannot define R_k until we have identified v_k^0 and v_k^n satisfying (3).

2.1.1. An example of a rank ordered pattern search method. In Fig. 1 we give an example of a rank ordered pattern search method. This algorithm is a sequential variant of the parallel multidirectional search algorithm of Dennis and Torczon [5].

One of three possible steps are accepted at the conclusion of each iteration: a *reflection* step, an *expansion* step, or a *shrink* step. Examples for each of these three steps in \mathbf{R}^2 can be seen in Figs. 2, 3, and 4, respectively. Note that since we know the

```

Start with an initial simplex with vertices  $\{v_0^0, v_0^1, \dots, v_0^n\}$ 
and evaluate  $f(v_0^j)$ ,  $j = 0, \dots, n$ .
for  $k = 0, 1, \dots$ 
  Reorder the vertices of the simplex so that
     $f(v_k^0) \leq f(v_k^1), \dots, f(v_k^{n-1}) \leq f(v_k^n)$ 
  Check the stopping criteria.
   $r_k \leftarrow 2v_k^0 - v_k^n$  /* reflection step */
  evaluate  $f(r_k)$ 
  if  $(f(r_k) < f(v_k^0))$  then
     $e_k \leftarrow 3v_k^0 - 2v_k^n$  /* expansion step */
    evaluate  $f(e_k)$ 
    if  $(f(e_k) < f(r_k))$  then
       $v_{k+1}^n \leftarrow e_k$  /* accept expansion */
      for  $j = 1, \dots, n - 1$  /* expand simplex */
         $v_{k+1}^j \leftarrow 3v_k^0 - 2v_k^j$ 
        evaluate  $f(v_{k+1}^j)$ 
      end for
    else
       $v_{k+1}^n \leftarrow r_k$  /* accept reflection */
      for  $j = 1, \dots, n - 1$  /* reflect simplex */
         $v_{k+1}^j \leftarrow 2v_k^0 - v_k^j$ 
        evaluate  $f(v_{k+1}^j)$ 
      end for
    end if
  else
    for  $j = 1, \dots, n$  /* shrink simplex */
       $v_{k+1}^j \leftarrow \frac{1}{2}v_k^0 + \frac{1}{2}v_k^j$ 
      evaluate  $f(v_{k+1}^j)$ 
    end
  end if
end

```

FIG. 1. A straightforward rank ordered pattern search method.

values of $f(v_k^0), \dots, f(v_k^n)$ upon entry into iteration k , this fairly conservative variant of a rank ordered pattern search method computes exactly $n + 1$ objective values at each iteration, regardless of the step that is finally accepted. More elaborate variants, which attempt to compute a single objective value per iteration, are possible. The variant we present here has the advantage of being both simple and robust.

2.2. Positive Basis Pattern Search Methods. The positive basis pattern search methods will be described in terms of the notion of positive linear dependence developed in [3]. Positive linear dependence captures the essential technical role played by the core pattern in [12].

The positive basis pattern search methods are also motivated by the requirements of parallel computing. As we have seen, the rank ordered pattern search methods may introduce a sequential element into the computation since we must satisfy (4). As we have seen, this may require the identification of the worst vertex before we can append the remaining search direction in the pattern. As a consequence, Amdahl's Law says

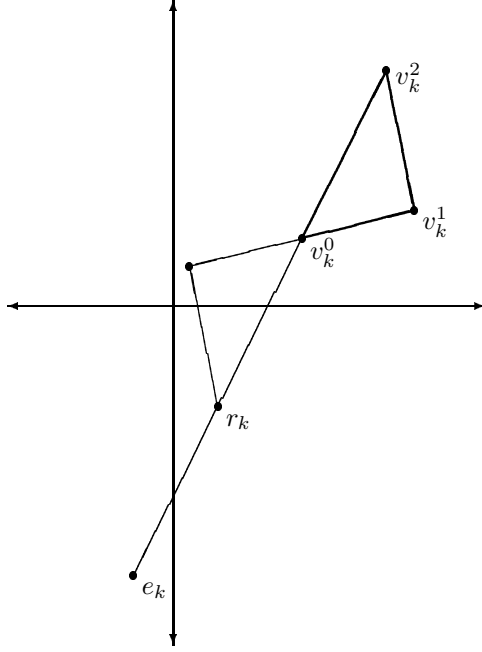


FIG. 2. A reflection step of the straightforward rank ordered pattern search method, given the simplex with vertices $\{v_k^0, v_k^1, v_k^2\}$.

that we can at most halve the execution time of a single iteration via computational parallelism, regardless of the number of processing units available. The positive basis pattern search methods avoid this sequentiality by imposing a geometric condition on Γ_k . Thus the positive basis pattern search methods are ideally suited for parallel implementation since the algebraic conditions on Γ_k can be satisfied *a priori* and allow all the necessary objective evaluations for a single iteration to be independently computed in parallel.

2.2.1. Positive linear dependence. We present here the ideas we will need from the theory of positive linear dependence [3]. The *positive span* of a set of vectors $\{a_1, \dots, a_r\}$ is the cone

$$\{ a \in \mathbf{R}^n \mid a = c_1 a_1 + \dots + c_r a_r, c_i \geq 0 \text{ for all } i \}.$$

The set $\{a_1, \dots, a_r\}$ 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 \mathbf{R}^n . The following theorem from [3] indicates that a positive spanning set contains at least $n + 1$ vectors.

THEOREM 2.1. *Suppose $\{a_1, \dots, a_r\}$ positively spans \mathbf{R}^n . Then $\{a_2, \dots, a_r\}$ linearly spans \mathbf{R}^n .*

The following characterizations of positive spanning sets can be found in [3] as well.

THEOREM 2.2. *Suppose $\{a_1, \dots, a_r\}$, $a_i \neq 0$, linearly spans \mathbf{R}^n . Then the following are equivalent:*

1. $\{a_1, \dots, a_r\}$ positively spans \mathbf{R}^n .
2. For every $b \neq 0$, there exists an i for which $b^T a_i > 0$.
3. For every $i = 1, \dots, r$, $-a_i$ is in the convex cone positively spanned by the remaining a_i .

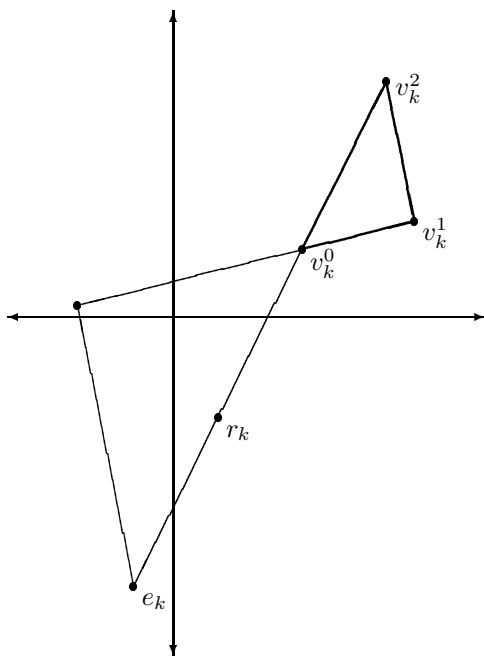


FIG. 3. An expansion step of the straightforward rank ordered pattern search method, given the simplex with vertices $\{v_k^0, v_k^1, v_k^2\}$.

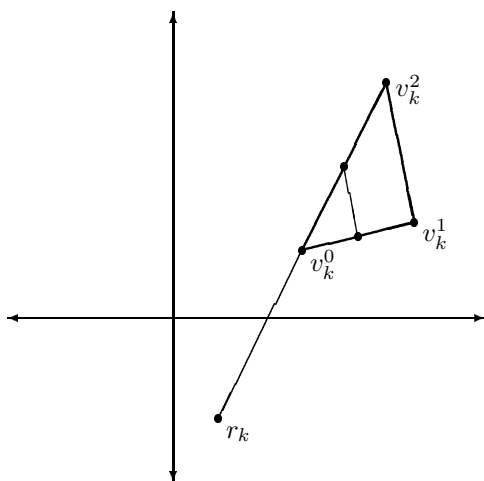


FIG. 4. A shrink step of the straightforward rank ordered pattern search method, given the simplex with vertices $\{v_k^0, v_k^1, v_k^2\}$.

A positive basis that contains $n + 1$ elements is called *minimal*. One can also show [3] that a positive basis can have no more than $2n$ elements; such a basis is called *maximal*. A maximal positive basis has a very special structure: it must consist of a linear basis for \mathbf{R}^n and the negatives of those basis vectors. On the other hand, it is easy to see that such a collection of $2n$ vectors is a positive basis, as the next two propositions show.

PROPOSITION 2.3. *Suppose $\{a_1, \dots, a_r\}$ is a positive basis for \mathbf{R}^n , and B is a nonsingular $n \times n$ matrix. Then the set $\{Ba_1, \dots, Ba_r\}$ is also a positive basis for \mathbf{R}^n .*

Proof. Because B is nonsingular, the set $\{Ba_1, \dots, Ba_r\}$ linearly spans \mathbf{R}^n . Moreover, because $\{a_1, \dots, a_r\}$ is a positive basis, by Theorem 2.2, part (2), given any $b \in \mathbf{R}^n$, $b \neq 0$, there exists an i for which

$$(B^T b)^T a_i = b^T B a_i > 0.$$

But this means that $\{Ba_1, \dots, Ba_r\}$ is a positive basis for \mathbf{R}^n . \square

PROPOSITION 2.4. *If B is a nonsingular $n \times n$ matrix, then the columns of the matrix $[B \ -B]$ form a positive basis.*

Proof. By part (2) of Theorem 2.2, we know that the matrix $[I \ -I]$ is a positive basis. The result then follows from Proposition 2.3. \square

2.2.2. Description of the positive basis pattern search algorithm. In a positive basis pattern search method, we require the generating matrix C_k to be any $n \times p_k$ matrix, $p_k > n + 1$, of the form found in (1), but now we require Γ_k to be a positive basis for \mathbf{R}^n . Proposition 2.3 then says that at every iteration k , $\Delta_k B \Gamma_k$ will also be a positive basis for \mathbf{R}^n .

2.2.3. Examples of positive basis pattern search methods. The classical pattern search methods considered in [12]—coordinate search with fixed step length, Hooke and Jeeves, and Evolutionary Operation using factorial designs—are all positive basis pattern search methods as we have defined them. The generating matrix in [12] has the form

$$C_k = [\ M_k \quad -M_k \quad L_k \quad 0 \]$$

for some $n \times n$ nonsingular matrix M_k . In light of the discussion in §2.2, the nature of $\Gamma_k = [M_k \ -M_k]$ as a maximal positive basis is now revealed.

We are not aware of any classical pattern search method that uses a positive basis that is not maximal. Coordinate search with fixed step length, Hooke and Jeeves, and Evolutionary Operation using factorial designs all use positive bases with $2n$ elements.

But it is not difficult to use the general notion of a positive basis to invent new pattern search methods that are proper extensions of the class of algorithms studied in [12]. Such methods have a practical appeal because we can reduce the number of objective evaluations in the worst case from $2n$ to as few as $n + 1$.

Here are two possible choices for Γ_k that lead to minimal positive basis pattern search methods. Let $e = (1, 1, \dots, 1)^T$, and consider

$$(6) \quad \Gamma_k = [nI \ -e]$$

and

$$(7) \quad \Gamma_k = [I \ -e].$$

It is not difficult to see from Theorem 2.2, part (2), that the columns of each matrix form a positive spanning set, and since there are $n + 1$ columns in each case these positive spanning sets are minimal positive bases.

Consider first the choice (6). If the basis matrix is given by $B = \frac{1}{n}[b_1 \ \dots \ b_n]$, then

$$B \Gamma_k = [b_1 \ \dots \ b_n \ b_{n+1}] \text{ where } b_{n+1} = -\frac{1}{n} \sum_{i=1}^n b_i.$$

Thus, the trial steps in this pattern are the vectors $\{b_1, \dots, b_n\}$ together with the negative of their average. In Fig. 5 we illustrate an example of such a minimal positive basis for \mathbf{R}^2 .

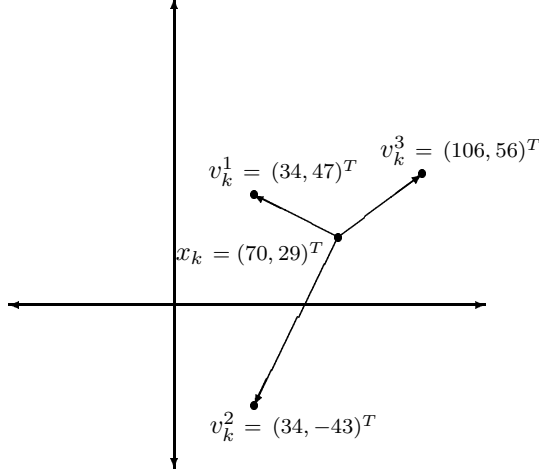


FIG. 5. A minimal positive basis for \mathbf{R}^2 with basis vectors $b_i = (v_k^i - x_k)$ for $i = 1, \dots, 3$.

For the choice (7), suppose $B = [b_1 \dots b_n]$. Then

$$B\Gamma_k = [b_1 \dots b_n \ b_{n+1}] \text{ where } b_{n+1} = -\sum_{i=1}^n b_i.$$

This pattern has an interpretation in terms of a simplex. We claim that these vectors are the vectors from the centroid of a simplex to the vertices of the simplex. If v_1, \dots, v_{n+1} are the vertices of a simplex and

$$\bar{v} = \frac{1}{n+1} \sum_{i=0}^n v^i$$

is the centroid, we wish to know whether we can find v_i for which $b_i = v_i - \bar{v}$. This is the same as seeking a solution of the system of equations

$$\tilde{V} \left(I - \frac{1}{n+1} ee^T \right) = \tilde{B},$$

where

$$\begin{aligned} \tilde{V} &= [v_1 \dots v_{n+1}] \\ \tilde{B} &= [b_1 \dots b_{n+1}]. \end{aligned}$$

Let

$$\tilde{A} = I - \frac{1}{n+1} ee^T;$$

then we are asking whether we can solve $\tilde{V}\tilde{A} = \tilde{B}$. We have the following alternative: either $\tilde{V}\tilde{A} = \tilde{B}$ has a solution, or there exists y such that $\tilde{A}y = 0, By \neq 0$. However, \tilde{A}

and \tilde{B} have the same nullspace: namely, that spanned by the vector e . Thus, it must be the case that we can solve $\tilde{V}\tilde{A} = \tilde{B}$, which means we can interpret the pattern as the vectors from the centroid of a simplex to the vertices of the simplex.

Having noted two possible positive bases that satisfy our requirements for a positive basis pattern search method, in Fig. 6 we give an example of a positive basis pattern search method. Note that this uses an exploratory moves strategy that is identical to that used for the Evolutionary Operation algorithm (see either [2], [10], or [12]). but instead of using a two-level factorial design, which requires 2^n objective values per iteration, the positive basis allows us to implement a design with as few as $n + 1$ objective values per iteration.

Start with an initial point x_0 , a positive basis $B\Gamma = [B\gamma^1 \cdots B\gamma^p]$, and $\Delta_0 > 0$.

Evaluate $f(x_0)$.

for $k = 0, 1, \dots$

 Check the stopping criteria.

for $i = 1, \dots, p$ /* this loop can be parallelized */

$s_k^i = \Delta_k B\gamma^i$

 evaluate $f(x_k + s_k^i)$

end for

$s_k = \operatorname{argmin}\{f(x_k + s_k^i)\}$ /* the sole synchronization point */

if $f(x_k + s_k) < f(x_k)$ **then**

$x_{k+1} = x_k + s_k$

$\Delta_{k+1} = \Delta_k$

else

$x_{k+1} = x_k$

$\Delta_{k+1} = \frac{1}{2}\Delta_k$

end if

end

FIG. 6. A straightforward positive basis pattern search method.

2.2.4. The relationship between positive basis and rank ordered pattern search methods. There is an overlap between the class of positive basis pattern search methods and rank ordered pattern search methods, as we shall soon demonstrate. However, it is instructive to see that the two classes are not equivalent.

To see that we cannot necessarily cast a rank ordered pattern search method as a positive basis pattern search method, we return to the example given in §2.1.1, where $(r_k - v_k^0) = (v_k^0 - v_k^2)$. Note that r_k is a legitimate reflection step for a rank ordered pattern search method since it satisfies (4) and (5). Fig. 7 shows that the set $\{(v_k^1 - v_k^0), (v_k^2 - v_k^0), (r_k - v_k^0)\}$ does not form a positive basis for \mathbf{R}^2 because we cannot find a member of the set for which the inner product with the vector $(b - v_k^0)$ is strictly positive. Thus we violate part (2) of Theorem 2.2.

To see that it is not always possible to cast a positive basis pattern search method as a rank ordered pattern search method consider the example shown in Fig. 8. Here the set of vectors $\{(r_k - v_k^0), (v_k^1 - v_k^0), (v_k^2 - v_k^0)\}$ does form a positive basis for \mathbf{R}^2 , but the vector $(r_k - v_k^0)$ does not constitute an acceptable reflection step for a rank ordered pattern search method. The difficulty in this instance is that $(r_k - v_k^0)$ is orthogonal to $(v_k^2 - v_k^0)$, thus violating conditions (4) and (5) which our analysis imposes upon rank ordered pattern search methods.

As can be seen in Fig. 9, the multidirectional search (MDS) algorithm lies in the

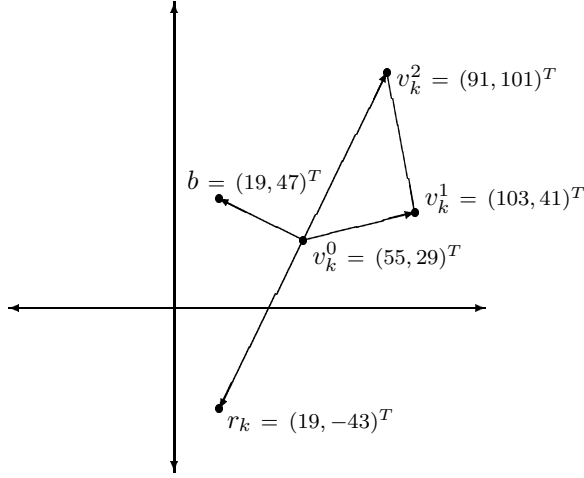


FIG. 7. A demonstration that a rank ordered pattern search method may not necessarily be a positive basis pattern search method

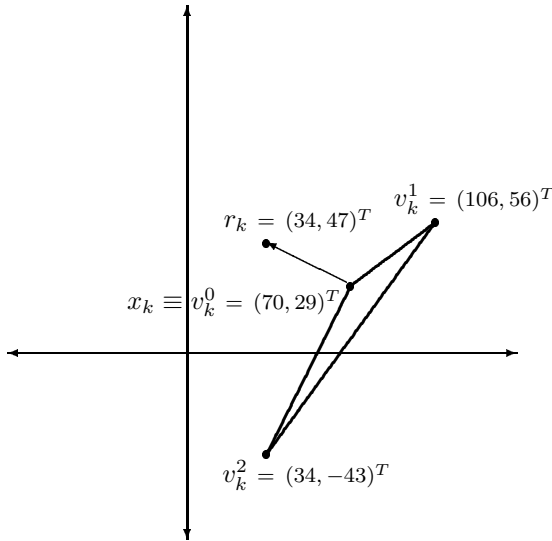


FIG. 8. A demonstration that a positive basis pattern search method may not necessarily be a rank ordered pattern search method

intersection of the two approaches. The set of vectors $\{(v_k^1 - v_k^0), \dots, (v_k^n - v_k^0), (v_k^0 - v_k^1), \dots, (v_k^0 - v_k^n)\} \equiv \{(v_k^1 - v_k^0), \dots, (v_k^n - v_k^0), (r_k^1 - v_k^0), \dots, (r_k^n - v_k^0)\}$ forms a maximal positive basis for \mathbf{R}^n . We can implement MDS as a parallel algorithm by simultaneously computing the objective values at all $2n$ points defined by the simplex and its reflection. In doing so, *a priori* we treat all the vertices of the original simplex, except the best vertex v_k^0 , as possibly being the worst and thus remove the sequential bottleneck or any need to coordinate such information across processes. This is equivalent to implementing a positive basis pattern search method with a maximal positive basis. However, we can certainly also implement MDS as a sequential algorithm, as demonstrated in §2.1.1, and take full advantage of rank order information in an effort

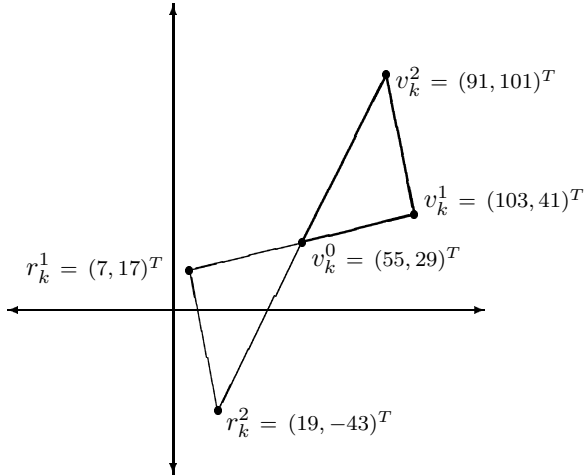


FIG. 9. A reflection step for a parallel implementation of the multidirectional search algorithm.

to reduce the number of function evaluations per iteration. We do so, however, at the expense of a sequential element that limits the effective use of parallel computing.

We close by noting that the convergence analysis for pattern search methods is flexible enough to accommodate a myriad of other combinations that may perhaps be more appropriate for a given application. We have illustrated only a few of the many options.

3. Convergence analysis. The convergence results for positive basis and rank ordered pattern search methods are like those in [12]:

THEOREM 3.1. *Suppose $L(x_0)$ is compact and suppose f is continuously differentiable on an open neighborhood Ω of $L(x_0)$. Let $\{x_k\}$ be the sequence of iterates produced by either a rank ordered or a positive basis pattern search method for unconstrained minimization (Algorithm 1).*

Then

$$\liminf_{k \rightarrow +\infty} \|g(x_k)\| = 0.$$

We will henceforth assume that f is continuously differentiable on an open set Ω containing the compact set $L(x_0)$. However, for Theorem 3.1 (and Theorem 3.2 to follow) we really only need f to be continuously differentiable on $L(x_0)$. Under that assumption, one would need first to derive estimates that show that if x_0 is not a stationary point then a pattern search method eventually will move into the interior of $L(x_0)$. One would then derive a similar set of estimates and show that these estimates are uniform on $L(x_1)$, in the interior of $L(x_0)$. The assumption that f is continuously differentiable on a set slightly larger than $L(x_0)$ makes the proof shorter and clearer at little expense of generality.

As in [12], we can also obtain a stronger result. To do so, we must ultimately stop allowing Δ_k to become larger, and we must require that the algorithm examine all the points of the core pattern, according to the Strong Hypotheses on Exploratory Moves.

With these restrictions, we obtain the following result:

Strong Hypotheses on Exploratory Moves.

1. $s_k \in \Delta_k P_k \equiv \Delta_k B C_k \equiv \Delta_k [B \Gamma_k B L_k 0]$.
2. If $\min\{f(x_k + y), y \in \Delta_k B \Gamma_k\} < f(x_k)$, then $f(x_k + s_k) \leq \min\{f(x_k + y), y \in \Delta_k B \Gamma_k\}$.

THEOREM 3.2. *Suppose $L(x_0)$ is compact and suppose f is continuously differentiable on an open neighborhood Ω of $L(x_0)$. In addition, assume that the columns of the generating matrices are uniformly bounded in norm, that in the update of Δ_k , we have $\lambda_k = 1$ for all k after some iteration, and that the pattern search method for unconstrained minimization (Algorithm 1) enforces the Strong Hypotheses on Exploratory Moves. Finally, let $\{x_k\}$ be the sequence of iterates produced by either a rank ordered or a positive basis pattern search method for unconstrained minimization.*

Then

$$\lim_{k \rightarrow +\infty} \|g(x_k)\| = 0.$$

By restricting the update of Δ_k to allow only $\lambda_k = 1$ for all k after some iteration, we are assured that $\lim_{k \rightarrow +\infty} \Delta_k = 0$. This is a corollary of Theorem 3.18.

The outline of the proof for Theorem 3.1 follows that of [12] as developed in [7]:

1. First we show that given $\eta > 0$, there exists $\delta > 0$ (independent of k) such that if $\|g(x_k)\| > \eta$ and $\Delta_k < \delta$, then a pattern search algorithm will find an acceptable step without further decrease of Δ_k .
2. We then show that if $\liminf_{k \rightarrow +\infty} \|g(x_k)\| \neq 0$, then there exists a nonzero lower bound on Δ_k .
3. Using purely algebraic properties of iterates produced by pattern search methods, we show that we must necessarily have $\liminf_{k \rightarrow +\infty} \Delta_k = 0$.
4. We then conclude that we must have $\liminf_{k \rightarrow +\infty} \|g(x_k)\| = 0$.

The analysis of positive basis pattern search methods and that of rank ordered pattern search methods differ only in the first step. After that the two lines of analysis converge and are more or less identical to similar steps in [12].

The sections that follow correspond to the outline given above. The new results can be found in §3.1, where we develop the case first for the positive basis pattern search methods and then for the rank ordered pattern search methods. The remaining sections complete the analysis, relying largely on results developed in [7] and [12].

3.1. Existence of a direction of descent. The following proposition is the justification for calling pattern search algorithms “methods of steep descent.”

PROPOSITION 3.3. *There exists $c > 0$ such that given any $\eta > 0$, we can find $\nu > 0$ such that if $\|g(x_k)\| > \eta$ and $\Delta_k < \nu$, there is a step $s_k^i \in \Delta_k B \Gamma_k$ for which*

$$-g_k^T s_k^i \geq c \|g_k\| \|s_k^i\|.$$

The proof of this result differs for positive basis pattern search methods and rank ordered pattern search methods and thus will be developed independently for each.

In both cases recourse will be had to the following result, the proof of which can be extracted from §6.2 of [12].

PROPOSITION 3.4. *Suppose that $\{a_1, \dots, a_n\}$ linearly spans \mathbf{R}^n . Then given any $x \in \mathbf{R}^n$, we can find an a_i for which*

$$|x^T a_i| \geq \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|,$$

where $\kappa(A)$ is the condition number of the matrix $A = [a_1 \cdots a_n]$.

We will also need the following proposition, which says that we can uniformly bound the first-order Taylor series remainder.

PROPOSITION 3.5. *Given $\varepsilon > 0$, we can find $\delta > 0$ such that if $x \in L(x_0)$ and $\|y - x\| \leq \delta$, then*

$$\left| f(y) - f(x) - g(x)^T(y - x) \right| \leq \varepsilon \|y - x\|.$$

Proof. Let $\delta_1 = \frac{1}{2} \min(1, \text{dist}(\partial L(x_0), \partial \Omega))$. If $x \in L(x_0)$ then the ball

$$B(x, \delta_1) = \{y \mid \|y - x\| < \delta_1\}$$

is contained in Ω . Let K be the closure of $\cup_{x \in L(x_0)} B(x, \delta_1)$; note the K is compact by construction.

Now, if $x \in L(x_0)$ and $y \in B(x, \delta_1)$ we may apply the mean-value theorem to obtain

$$f(y) - f(x) = g(z)^T(y - x)$$

for some z on the line segment connecting x and y . Then

$$\begin{aligned} \left| f(y) - f(x) - g(x)^T(y - x) \right| &= \left| (g(z) - g(x))^T(y - x) \right| \\ &\leq \|g(z) - g(x)\| \|y - x\| \end{aligned}$$

The uniform continuity of g on K allows us to find $\delta_2 > 0$ such that if $x, y \in K$ and $\|y - x\| < \delta_2$ then

$$\|g(z) - g(x)\| < \varepsilon.$$

Thus, if we choose $\delta = \min(\delta_1, \delta_2)$, then if $x \in L(x_0)$ and $\|y - x\| < \delta$, we have

$$\left| f(y) - f(x) - g(x)^T(y - x) \right| \leq \varepsilon \|y - x\|.$$

□

3.1.1. The case of positive basis pattern search methods. For positive basis pattern search methods, Proposition 3.3 follows from purely geometric properties of the pattern without reference to the objective and the requirement that Δ_k be sufficiently small.

We begin by generalizing Proposition 3.4 to positive spanning sets.

PROPOSITION 3.6. *Given any set $\{a_1, \dots, a_r\}$ that positively spans \mathbf{R}^n , $a_i \neq 0$ for $i = 1, \dots, r$, there exists $c > 0$ such that for all $x \in \mathbf{R}^n$, we can find an a_i for which*

$$x^T a_i \geq c \|x\| \|a_i\|.$$

Proof. We need consider only the case $x \neq 0$. According to Theorem 2.1, we can find a basis for \mathbf{R}^n among the vectors $\{a_1, \dots, a_r\}$. Thus, possibly upon reordering, we may assume that the matrix

$$A = [a_1 \cdots a_n]$$

is invertible. Then by Proposition 3.4, given any $x \in \mathbf{R}^n$ we can find a column a_i in A for which

$$(8) \quad |x^T a_i| \geq \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|.$$

There are two possibilities to consider in (8). The first is that

$$(9) \quad x^T a_i \geq \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|;$$

there is nothing more to be done in this case.

The other possibility is that

$$-x^T a_i \geq \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|.$$

In this case, we first appeal to part (3) of Theorem 2.2 to choose a set of scalars $\mu_{jk} \geq 0$, depending only on $\{a_1, \dots, a_r\}$, such that for any j we can express $-a_j$ as

$$-a_j = \sum_{k=1, k \neq j}^r \mu_{jk} a_k.$$

This representation for $-a_i$ leads to

$$\sum_{k=1, k \neq i}^r \mu_{ik} x^T a_k \geq \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|.$$

It must then be the case that for some index $\ell \neq i$ we have

$$\mu_{i\ell} x^T a_\ell \geq \frac{1}{r-1} \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|.$$

Since $x \neq 0$ we know that it must be the case that $\mu_{i\ell} > 0$, so we arrive at

$$x^T a_\ell \geq \frac{1}{\mu_{i\ell}} \frac{1}{r-1} \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_i\|.$$

If we let

$$\mu^* = \max \{ \mu_{jk} \mid 1 \leq j, k \leq r, j \neq k \}$$

and

$$a_* = \min \left\{ \frac{\|a_j\|}{\|a_k\|} \mid 1 \leq j, k \leq r \right\},$$

we obtain

$$(10) \quad x^T a_\ell \geq \frac{a_*}{\mu^*} \frac{1}{r-1} \frac{1}{\kappa(A)\sqrt{n}} \|x\| \|a_\ell\|.$$

Combining (9) and (10) then yields the proposition. \square

Proposition 3.3 in the case of positive basis pattern search methods now follows from the following stronger result.

PROPOSITION 3.7. For a positive basis pattern search method there exists $c > 0$ such that for any k , there is a step $s_k^i \in \Delta_k B\Gamma_k$ for which

$$-g_k^T s_k^i \geq c \|g_k\| \|s_k^i\|.$$

Proof. At each iteration k , $B\Gamma_k$ is a positive basis. Thus, by Proposition 3.6, there exists $C(B\Gamma_k) > 0$ for which we can find $v \in B\Gamma_k$ such that

$$-g_k^T v \geq C(B\Gamma_k) \|g_k\| \|v\|.$$

However, Γ_k is a member of the finite set of matrices \mathbf{M} , so there are only finitely many possibilities for $B\Gamma_k$. Taking

$$c = \min_{\Gamma_k \in \mathbf{M}} C(B\Gamma_k) > 0$$

and multiplying v by Δ_k yields the proposition. \square

3.1.2. The case of rank ordered pattern search methods. For rank ordered pattern search methods, we are only assured of a suitable descent direction when Δ_k is sufficiently small. When Δ_k is sufficiently small, the relative ranking of the extreme objective values (best and worst or lowest and highest) determines a direction that is suitably close to the direction of steepest descent.

Because the matrices Γ_k come from a finite set of matrices, the relative sizes of steps in the core pattern remain bounded and are related to Δ_k :

PROPOSITION 3.8. There exist $r, R > 0$ such that for all k , if $s_k^i \in \Delta_k B\Gamma_k$, then $r\Delta_k \leq \|s_k^i\| \leq R\Delta_k$.

Consequently, given $\varepsilon > 0$, there exists $\delta > 0$, independent of k , such that if $\Delta_k < \delta$, then $\|s_k^i\| \leq \varepsilon$ for all $s_k^i \in \Delta_k B\Gamma_k$.

Proof. The restrictions on Γ_k for both the rank ordered and the positive basis pattern search methods ensure that none of the columns of Γ_k can be zero. Since B is invertible, $B\Gamma_k$ can never have a zero column. Because $\Gamma_k \in \mathbf{M}$ and \mathbf{M} is a finite set of matrices it follows that there exist a nonzero lower bound and a finite upper bound for the norm of all the columns of elements of $B\mathbf{M}$. The proposition follows. \square

If we let $\gamma_* = r/R$ and $\gamma^* = R/r$, then Proposition 3.8 means that for any $s_k^i, s_k^j \in \Delta_k B\Gamma_k$, we have

$$(11) \quad \gamma_* \|s_k^j\| \leq \|s_k^i\| \leq \gamma^* \|s_k^j\|.$$

Proposition 3.9 translates the rank ordering of the vertices into a statement about directions of descent. It says that when Δ_k is small enough, the edges from the vertices v_k^1, \dots, v_k^n to the best vertex v_k^0 are either descent directions or are not very steep ascent directions.

PROPOSITION 3.9. Given $\varepsilon > 0$ and $\eta > 0$, there exists $\nu > 0$ such that if $\|g(v_k^0)\| > \eta$ and $\Delta_k < \nu$, then

$$g(v_k^0)^T (v_k^0 - v_k^i) \leq \varepsilon \|g(v_k^0)\| \|v_k^0 - v_k^i\| \quad i = 1, \dots, n.$$

Proof. We have $f(v_k^0) - f(v_k^i) \leq 0$ for all $i = 1, \dots, n$. Applying Propositions 3.5 and 3.8 we can find $\nu > 0$ such that if $\Delta_k < \nu$, then

$$g(v_k^0)^T (v_k^0 - v_k^i) - \varepsilon \eta \|v_k^0 - v_k^i\| \leq f(v_k^0) - f(v_k^i) \leq 0.$$

The result follows. \square

The next step is to show that when Δ_k is sufficiently small, the direction from the worst vertex to the best vertex is a direction of steep descent from v_k^0 .

PROPOSITION 3.10. *There exists $c > 0$ such that given any $\eta > 0$, we can find $\nu > 0$ such that if $\|g(v_k^0)\| > \eta$ and $\Delta_k < \nu$, then*

$$-g(v_k^0)^T(v_k^0 - v_k^n) \geq c \|g(v_k^0)\| \|v_k^0 - v_k^n\|.$$

Proof. Let

$$\kappa^* = \max_{S_k \subset \Gamma_k \in \mathbf{M}} \kappa(BS_k),$$

where by $S_k \subset \Gamma_k$ we refer to the partition $\Gamma_k = [S_k \ R_k]$. By Proposition 3.4, there is some index ℓ , $1 \leq \ell \leq n$, for which

$$|g(v_k^0)^T(v_k^0 - v_k^\ell)| \geq \frac{1}{\kappa^* \sqrt{n}} \|g(v_k^0)\| \|v_k^0 - v_k^\ell\|.$$

If Δ_k is sufficiently small, we can divine the sign of the inner product: by Proposition 3.9, we can find $\nu_1 > 0$ such that if $\Delta_k < \nu_1$, then

$$g(v_k^0)^T(v_k^0 - v_k^\ell) \leq \frac{1}{2} \frac{1}{\kappa^* \sqrt{n}} \|g(v_k^0)\| \|v_k^0 - v_k^\ell\|.$$

This means that we must actually have

$$(12) \quad -g(v_k^0)^T(v_k^0 - v_k^\ell) \geq \frac{1}{\kappa^* \sqrt{n}} \|g(v_k^0)\| \|v_k^0 - v_k^\ell\|.$$

This shows the existence of a good descent direction. Now we relate (12) to the distinguished direction $v_k^0 - v_k^n$. Because $f(v_k^i) \leq f(v_k^n)$ for all i ,

$$f(v_k^0) - f(v_k^n) \leq f(v_k^0) - f(v_k^\ell).$$

Using this inequality and Propositions 3.5 and 3.8, given any $\varepsilon > 0$ we can find $\nu_2 > 0$ such that if $\Delta_k < \nu_2$ then

$$\begin{aligned} g(v_k^0)^T(v_k^0 - v_k^n) - \varepsilon \|v_k^0 - v_k^n\| &\leq f(v_k^0) - f(v_k^n) \\ &\leq f(v_k^0) - f(v_k^\ell) \leq g(v_k^0)^T(v_k^0 - v_k^\ell) + \varepsilon \|v_k^0 - v_k^\ell\|, \end{aligned}$$

or

$$(13) \quad -g(v_k^0)^T(v_k^0 - v_k^n) \geq -g(v_k^0)^T(v_k^0 - v_k^\ell) - \varepsilon (\|v_k^0 - v_k^\ell\| + \|v_k^0 - v_k^n\|).$$

Now choose

$$\varepsilon = \frac{1}{4} \frac{1}{\kappa^* \sqrt{n}} \eta \min(1, \gamma_*),$$

and ν_2 accordingly, where γ_* is as in (11). Let $\nu = \min(\nu_1, \nu_2)$. Then, if $\Delta_k < \nu$, (12) holds. Meanwhile,

$$(14) \quad \begin{aligned} \varepsilon \|v_k^0 - v_k^\ell\| &\leq \frac{1}{4} \frac{1}{\kappa^* \sqrt{n}} \eta \|v_k^0 - v_k^\ell\| \\ \varepsilon \|v_k^0 - v_k^n\| &\leq \frac{1}{4} \frac{1}{\kappa^* \sqrt{n}} \eta \gamma_* \|v_k^0 - v_k^n\| \leq \frac{1}{4} \frac{1}{\kappa^* \sqrt{n}} \eta \|v_k^0 - v_k^\ell\|. \end{aligned}$$

Then (12), (13), and (14) yield

$$-g(v_k^0)^T(v_k^0 - v_k^n) \geq \frac{1}{2} \frac{1}{\kappa^* \sqrt{n}} \|g(v_k^0)\| \|v_k^0 - v_k^\ell\|.$$

Applying (11) again yields

$$-g(v_k^0)^T(v_k^0 - v_k^n) \geq \gamma_* \frac{1}{2} \frac{1}{\kappa^* \sqrt{n}} \|g(v_k^0)\| \|v_k^0 - v_k^n\|,$$

which is the desired estimate. \square

We can now attend to the reflection step σ_k from (5), which may comprise all the directions to the best vertex from the other vertices.

PROPOSITION 3.11. *There exists $c > 0$ such that given any $\eta > 0$, we can find $\nu > 0$ such that if $\|g(v_k^0)\| > \eta$ and $\Delta_k < \nu$, then*

$$-g(v_k^0)^T \sigma_k \geq c \|g(v_k^0)\| \|\sigma_k\|.$$

Proof. Recall that the reflection step has the form

$$\sigma_k = \sum_{i=1}^n \rho_k^i (v_k^0 - v_k^i), \quad \rho_k^i \geq 0, \quad \rho_k^n > 0,$$

so

$$\begin{aligned} -g(v_k^0)^T \sigma_k &= -\sum_{i=1}^n \rho_k^i g(v_k^0)^T (v_k^0 - v_k^i) \\ (15) \quad &= -\rho_k^n g(v_k^0)^T (v_k^0 - v_k^n) - \sum_{i=1}^{n-1} \rho_k^i g(v_k^0)^T (v_k^0 - v_k^i). \end{aligned}$$

Using Proposition 3.10 we can find $\nu_1 > 0$ such that if $\Delta_k < \nu_1$, then

$$(16) \quad -g(v_k^0)^T (v_k^0 - v_k^n) \geq C \|g(v_k^0)\| \|v_k^0 - v_k^n\|.$$

Meanwhile, given any $\varepsilon > 0$, Proposition 3.9 allows us to find ν_2 so that if $\Delta_k \leq \nu_2$, then

$$g(v_k^0)^T (v_k^0 - v_k^i) \leq \varepsilon \|g(v_k^0)\| \|v_k^0 - v_k^i\|$$

for all $i = 1, \dots, n-1$. We will choose ε felicitously in a moment. In the meantime, applying (11) to the previous bound yields

$$(17) \quad g(v_k^0)^T (v_k^0 - v_k^i) \leq \varepsilon \gamma^* \|g(v_k^0)\| \|v_k^0 - v_k^n\|.$$

Now let $\nu = \min(\nu_1, \nu_2)$ and suppose $\Delta_k \leq \nu$. Returning to (15), we can apply (16) and (17) to obtain the bound

$$\begin{aligned} -g(v_k^0)^T \sigma_k &\geq C \rho_k^n \|g(v_k^0)\| \|v_k^0 - v_k^n\| - \sum_{i=1}^{n-1} \rho_k^i \varepsilon \gamma^* \|g(v_k^0)\| \|v_k^0 - v_k^n\| \\ &\geq \left(C \rho_k^n - \varepsilon \gamma^* \sum_{i=1}^{n-1} \rho_k^i \right) \|g(v_k^0)\| \|v_k^0 - v_k^n\|. \end{aligned}$$

Now we choose $\varepsilon > 0$. Let

$$\rho_* = \min_k \rho_k^n$$

and

$$\varepsilon < \frac{C}{2} \frac{1}{\gamma^*} \min_k \frac{\rho_*}{\sum_{i=1}^{n-1} \rho_k^i}.$$

We are assured of a nonzero ρ_* and ε because $\Gamma_k = [S_k \ R_k] \in \mathbf{M}$, and \mathbf{M} is a finite set of matrices, so the minimum in each of the two preceding relations is taken only over a finite set. With this choice of ε and ρ_* , we obtain

$$(18) \quad -g(v_k^0)^T \sigma_k > \frac{C}{2} \rho_* \|g(v_k^0)\| \|v_k^0 - v_k^n\|.$$

Finally, we must relate the norm of $v_k^0 - v_k^n$ to that of σ_k . We have

$$\sigma_k = \sum_{i=1}^n \rho_k^i (v_k^0 - v_k^i)$$

with $\rho_k^i \geq 0$, so

$$(19) \quad \|\sigma_k\| \leq \sum_{i=1}^n \rho_k^i \|v_k^0 - v_k^i\|$$

Now, the vectors $v_k^0 - v_k^i, i = 1, \dots, n$ are the columns of $\Delta_k B S_k$, possibly permuted:

$$[(v_k^0 - v_k^1) \cdots (v_k^0 - v_k^n)] = \Delta_k B S_k \Pi_k,$$

where Π_k is a permutation matrix. Consider any invertible $n \times n$ matrix A with columns a_1, \dots, a_n . For any i, j , from $e_j = A^{-1} a_j$ and $a_i = A e_i$ we obtain the inequalities $1 \leq \|A^{-1}\| \|a_j\|$ and $\|a_i\| \leq \|A\|$, whence $\|a_i\| \leq \kappa(A) \|a_j\|$. Thus for any i we have

$$\|v_k^0 - v_k^i\| \leq \kappa(\Delta_k B S_k \Pi_k) \|v_k^0 - v_k^n\| = \kappa(B S_k) \|v_k^0 - v_k^n\|.$$

The latter equality holds because Π_k is an orthogonal transformation. Returning to (19), we then have

$$\|\sigma_k\| \leq \left(\rho_k^n + \sum_{i=1}^{n-1} \rho_k^i \kappa(B S_k) \right) \|v_k^0 - v_k^n\|.$$

Since there are only finitely many choices for S_k and the ρ_k^i , we can find a constant K , independent of k , such that

$$(20) \quad \|\sigma_k\| \leq K \|v_k^0 - v_k^n\|.$$

The result then follows from (18) and (20). \square

A consequence of the preceding proposition is that if v_k^0 is not a stationary point and if Δ_k is sufficiently small, then the reflection step will improve upon the best

objective value $f(v_k^0)$, and not just $f(x_k)$. However, we have chosen to pose Proposition 3.3 in terms of x_k , which is not necessarily v_k^0 . Proposition 3.3 now follows as a corollary of Proposition 3.11 and the uniform continuity of g :

COROLLARY 3.12. *There exists $c > 0$ such that given any $\eta > 0$, we can find $\nu > 0$ such that if $\|g(v_k^0)\| > \eta$ and $\Delta_k < \nu$, then*

$$-g(x_k)^T \sigma_k \geq c \|g(x_k)\| \|\sigma_k\|.$$

Proof. Choose $\nu_1 > 0$ so small that if $\Delta_k < \nu_1$, then the conclusion of Proposition 3.11 holds. Then

$$\begin{aligned} -g(x_k)^T \sigma_k &= -g(v_k^0)^T \sigma_k + (g(v_k^0) - g(x_k))^T \sigma_k \\ &\geq C \|g(v_k^0)\| \|\sigma_k\| - \|g(v_k^0) - g(x_k)\| \|\sigma_k\|. \end{aligned}$$

Next choose $\nu_2 > 0$ so small that if $\Delta_k < \nu_2$, then

$$\|g(v_k^0) - g(x_k)\| \leq \frac{C}{4} \eta$$

and

$$\|g(v_k^0)\| \geq \frac{1}{2} \|g(x_k)\|.$$

Then, if $\Delta_k < \min(\nu_1, \nu_2)$,

$$-g(x_k)^T \sigma_k \geq \frac{C}{4} \|g(x_k)\| \|\sigma_k\|.$$

□

3.2. Finding an acceptable step. From this point on the convergence analyses for positive basis pattern search methods and rank ordered pattern search methods are identical and follow more or less directly from results developed in [7] and [12].

The following two results come from [12], to which we refer the reader for the proofs. The first result indicates one sense in which Δ_k regulates step length.

LEMMA 3.13 (Lemma 3.1 from [12]). *There exists a constant $\zeta_* > 0$, independent of k , such that for any trial step $s_k^i \neq 0$ produced by a pattern search method for unconstrained minimization (Algorithm 1) we have $\|s_k^i\| \geq \zeta_* \Delta_k$.*

We also recall

LEMMA 3.14 (Lemma 3.6 from [12]). *If there exists a constant $C > 0$ such that for all k , $C > \|c_k^i\|$, for all $i = 1, \dots, p$, then there exists a constant $\psi_* > 0$, independent of k , such that for any trial step s_k^i produced by a pattern search method for unconstrained minimization (Algorithm 1) we have $\Delta_k \geq \psi_* \|s_k^i\|$.*

We are now ready to state and prove the main result for this section.

PROPOSITION 3.15. *Suppose that $L(x_0)$ is compact and that f is continuously differentiable on an open neighborhood Ω of $L(x_0)$. Then given any $\eta > 0$, there exists $\delta > 0$, independent of k , such that if $\Delta_k < \delta$ and $\|g(x_k)\| > \eta$, then either the positive basis or rank ordered pattern search method will find an acceptable step s_k ; i.e., $f(x_k + s_k) < f(x_k)$.*

If, in addition, the columns of the generating matrices remain bounded in norm and we enforce the Strong Hypotheses on Exploratory Moves, then, given any $\eta > 0$,

there exist $\delta > 0$ and $c > 0$, independent of k , such that if $\Delta_k < \delta$ and $\|g(x_k)\| > \eta$, then

$$f(x_{k+1}) \leq f(x_k) - c \|g(x_k)\| \|s_k\|.$$

Proof. Proposition 3.3 assures us that we can find $\delta_1 > 0$ such that if $\Delta_k < \delta_1$, then there is a step $s_k^i \in \Delta_k B\Gamma_k$ for which

$$(21) \quad g(x_k)^T s_k^i \leq -c \|g(x_k)\| \|s_k^i\|.$$

Meanwhile, Proposition 3.5 says that we can choose $\delta_2 > 0$ such that if $\Delta_k < \delta_2$, then

$$(22) \quad f(x_k + s_k^i) - f(x_k) \leq g(x_k)^T s_k^i + \frac{c}{2} \eta \|s_k^i\|.$$

Thus, if $\Delta_k < \delta = \min(\delta_1, \delta_2)$, (21) and (22) yield

$$f(x_k + s_k^i) - f(x_k) \leq -\frac{c}{2} \|g(x_k)\| \|s_k^i\|,$$

and so $f(x_k^i) \equiv f(x_k + s_k^i) < f(x_k)$ for at least one $s_k^i \in \Delta_k B\Gamma_k$. The Hypotheses on Exploratory Moves guarantee that if

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

then $f(x_k + s_k) < f(x_k)$. This proves the first part of the Proposition.

If, in addition, we enforce the Strong Hypotheses on Exploratory Moves, then we actually have

$$f(x_k + s_k) - f(x_k) \leq -\frac{c}{2} \|g_k\| \|s_k^i\|.$$

Lemma 3.13 then ensures that

$$f(x_k + s_k) \leq f(x_k) - \frac{c}{2} \zeta_* \Delta_k \|g(x_k)\|.$$

Applying Lemma 3.14, we arrive at

$$f(x_k + s_k) \leq f(x_k) - \frac{c}{2} \zeta_* \psi_* \|g(x_k)\| \|s_k\|,$$

which is the desired estimate. \square

The following corollary, which follows from Proposition 3.15 and the update for Δ_k , corresponds to Proposition 3.4 in [12]:

COROLLARY 3.16. *Suppose that $L(x_0)$ is compact and that f is continuously differentiable on an open neighborhood Ω of $L(x_0)$. If $\liminf_{k \rightarrow +\infty} \|g(x_k)\| \neq 0$, then there exists a constant $\Delta_* > 0$ such that for all k , $\Delta_k > \Delta_*$.*

If $\liminf_{k \rightarrow +\infty} \|g(x_k)\| \neq 0$, then Proposition 3.15 says that there is a uniform bound $\delta > 0$ such that once $\Delta_k < \delta$, the pattern search algorithm will necessarily find an acceptable step. Since we reduce Δ_k only if we have an unsuccessful iteration, this means we would at some point stop reducing Δ_k .

3.3. The algebraic nature of the iterates. The iterates produced by pattern search methods have a specific algebraic form. The rank ordered and positive basis pattern search methods inherit this basic algebraic structure. The next result, a proof of which can be found in [12], is key to the convergence of pattern search methods.

THEOREM 3.17 (Theorem 3.2 from [12]). *Any iterate x_N produced by a pattern search method for unconstrained minimization (Algorithm 1) can be expressed in the following form:*

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

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 algorithm for updating Δ_k (Algorithm 2),
- r_{LB} and r_{UB} depend on N ,
- Δ_0 is the initial choice for the step length control parameter,
- B is the basis matrix, and
- $z_k \in \mathbf{Z}^n$, $k = 0, \dots, N - 1$.

The next theorem combines the strict algebraic structure of the iterates, the simple decrease condition, and the algorithm for updating Δ_k , to reach a conclusion about the limiting behavior of Δ_k .

THEOREM 3.18. *Suppose that $L(x_0)$ is compact. Then $\liminf_{k \rightarrow +\infty} \Delta_k = 0$.*

The proof is identical to that of Theorem 3.3 in [12]. Briefly, Theorem 3.17 says that the iterates lie on a rational lattice. If $\liminf_{k \rightarrow +\infty} \Delta_k \neq 0$, then there can be only a finite number of distinct points visited by the algorithm. However, Proposition 3.15 says that unless we are at a stationary point, we will eventually choose a new iterate with a strictly lower objective value; this is at odds with the fact that we only visit a finite number of distinct points.

3.4. The proofs of Theorem 3.1 and Theorem 3.2. The conclusion of the proof of Theorem 3.1 is identical to that of Theorem 3.5 in [12]. Suppose that $\liminf_{k \rightarrow +\infty} \|g(x_k)\| \neq 0$. Then Corollary 3.16 tells us that there exists $\Delta_* > 0$ such that for all k , $\Delta_k \geq \Delta_*$. But this contradicts Theorem 3.18.

The proof of Theorem 3.2 follows that of Theorem 3.7 in [12], to which we refer the reader.

4. Concluding remarks. We believe that versions of the two classes of algorithms we have introduced in this paper can be developed for bound constrained minimization as well. The positive basis pattern search methods for unconstrained minimization require a positive basis for \mathbf{R}^n ; the correct analog for bound constrained minimization should require a positive basis for the tangent cone of the feasible region at each iterate. The work in [7] uses such a basis that appears to be maximal in size. The theory of positive linear dependence used in the unconstrained case suggests a line of development to sharpen the results in [7], which may in turn reduce the computational cost per iteration in the bound constrained case.

Similarly, the heuristic of approximating the direction of steepest descent using the best and worst of $n + 1$ vertices (determined by their objective values) should also be applicable to the bound constrained case. There we would only consider feasible points and develop a crude approximation for the direction considered in gradient projection algorithms.

REFERENCES

- [1] G. E. P. BOX, *Evolutionary operation: a method for increasing industrial productivity*, Applied Statistics, 6 (1957), pp. 81–101.
- [2] M. J. BOX, D. DAVIES, AND W. H. SWANN, *Non-Linear Optimization Techniques*, ICI Monograph No. 5, Oliver & Boyd, Edinburgh, 1969.
- [3] O. L. DAVIES, *The Design and Analysis of Industrial Experiments*, Hafner Publishing Company, New York, 1954.
- [4] J. E. DENNIS AND V. TORCZON, *Managing approximation models in optimization*, in Multidisciplinary Design Optimization: State-of-the-Art, N. M. Alexandrov and M. Y. Hussaini, eds., SIAM, Philadelphia, 1997.
- [5] J. E. DENNIS, JR. AND V. TORCZON, *Direct search methods on parallel machines*, SIAM Journal on Optimization, 1 (1991), pp. 448–474.
- [6] R. HOOKE AND T. A. JEEVES, *Direct search solution of numerical and statistical problems*, Journal of the Association for Computing Machinery (ACM), 8 (1961), pp. 212–229.
- [7] R. M. LEWIS AND V. J. TORCZON, *Pattern search algorithms for bound constrained minimization*, Tech. Rep. 96-20, ICASE, Mail Stop 403, NASA Langley Research Center, Hampton, Virginia 23681-0001, 1996. To appear in SIAM J. on Optimization.
- [8] K. I. M. MCKINNON, *Convergence of the Nelder–Mead simplex method to a non-stationary point*, Tech. Rep. MS 96-006, Department of Mathematics and Statistics, University of Edinburgh, May 1996. To appear in SIAM Journal on Optimization.
- [9] 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.
- [10] W. H. SWANN, *Direct search methods*, in Numerical Methods for Unconstrained Optimization, W. Murray, ed., Academic Press, London and New York, 1972, pp. 13–28.
- [11] V. TORCZON, *Multi-Directional Search: A Direct Search Algorithm for Parallel Machines*, PhD thesis, Department of Mathematical Sciences, Rice University, Houston, TX, 1989; available as TR90-07, Department of Computational and Applied Mathematics, Rice University, Houston, TX.
- [12] ———, *On the convergence of pattern search algorithms*, SIAM Journal on Optimization, 7 (1997), pp. 1–25.
- [13] Y. WEN-CI, *The convergence property of the simplex evolutionary techniques*, Scientia Sinica, Special Issue of Mathematics, 1 (1979), pp. 68–77.
- [14] ———, *Positive basis and a class of direct search techniques*, Scientia Sinica, Special Issue of Mathematics, 1 (1979), pp. 53–67.