Pattern Search Algorithms for Mixed Variable Programming

Charles Audet and J .E. Dennis, Jr.

CRPC-TR99875
Revised February 2000

Center for Research on Parallel Computation
Rice University
6100 South Main Street
CRPC - MS 41
Houston, TX 77005

Submitted February 1999

On the Convergence of Mixed Integer Pattern Search Algorithms

Charles Audet
J.E. Dennis Jr.

{charlesa, dennis}@caam.rice.edu
Rice University
Department of Computational
and Applied Mathematics
6100 South Main Street - MS 134
Houston, Texas, 77005-1892 USA

May 3, 1999

Abstract: The definition of pattern search methods for solving nonlinear unconstrained optimization problems is generalized here to include integer variables. The notion of local optimality in mixed integer programming is defined through a user-specified set of neighboring points. We present a generalized pattern search algorithm that provides an accumulation point that satisfies some necessary conditions for local optimality. This point is the limit of a subsequence of unsuccessful iterates whose corresponding mesh size parameters converge to zero. We present a stronger, more expensive, version of the algorithm that guarantees additional necessary optimality conditions. A small example illustrates the differences between the two versions of the algorithm.

Key words Pattern search algorithm, convergence analysis, unconstrained optimization, mixed integer programming.

Acknowledgments: Work of the first author was supported by NSERC (Natural Sciences and Engineering Research Council) fellowship PDF-207432-1998 and by CRPC (Center for Research on Parallel Computing). Work of the second author was supported by DOE DE-FG03-95ER25257, AFOSR F49620-98-1-0267, The Boeing Company, Sandia LG-4253, Mobil and CRPC CCR-9120008.

1 Introduction

Torczon [11] presented a general definition of an abstract pattern search method. The objective of the method is to minimize a continuously differentiable function $f : \mathrm{R}^n \to \mathrm{R}$ without any knowledge of its derivative. Torczon [11] shows that the method includes algorithms such as *coordinate search* with fixed step sizes, *evolutionary operation* using factorial design [2], the original *pattern search algorithm* [7], and the *multidirectional search algorithm* [6]. An achievement of [11] is to develop general convergence results for the entire class of the methods. A survey of derivative free methods for unconstrained optimization can be found in Conn, Scheinberg and Toint [4].

The main result of [11] is that under mild assumptions, the sequence of iterates (x_k) of R^n generated by any pattern search method satisfies

$$\liminf_{k\to\infty} \|\nabla f(x_k)\| \quad = \quad 0, \tag{1}$$

without ever computing or explicitly approximating derivatives. At each iteration, the function is evaluated at trial points on a discrete mesh surrounding the current iterate in search of one yielding decrease in the objective function value. Lewis and Torczon [9] use positive basis theory to strengthen the result by roughly cutting in half the worst case number of trial points at each iterations without affecting the convergence result. Lewis and Torczon [8] [10] extend pattern search algorithms and the convergence theory to bound and linearly constrained minimization.

The main objective of the present paper is to further generalize the problem to be solved because many engineering optimization problems contain both continuous and discrete variables.

We consider the problem of minimizing the function $f : \Omega \to \mathrm{R}$, where the domain is partitioned into continuous and discrete variables $\Omega = \mathrm{R}^{n^c} \times \mathrm{Z}^{n^d}$, and n^c and n^d are the dimensions of the corresponding spaces. The function f is assumed to be continuously differentiable when the variables in Z^{n^d} are fixed. We present a general pattern search method that reduces to that of Torczon [11] when the dimension n^d is fixed to zero. The iterates generated by the method are partitioned into continuous and discrete variables $x_k = (x_k^c, x_k^d)$, where $x_k^c \in \mathrm{R}^{n^c}$ and $x_k^d \in \mathrm{Z}^{n^d}$.

A second objective of the paper is to slightly generalize the part of the algorithm that deals with the continuous variables and to revise and shorten the arguments developed in [11] and in [9]. We first show how to obtain an accumulation point $\hat{x}$ of the sequence of iterates that satisfies first-order optimality conditions with respect to the continuous variables. These conditions reduce to (1) when there are no discrete variables. We also guarantee that the same limit point $\hat{x}$ satisfies some local optimality conditions with respect to the discrete variables. The notion of local optimality is defined through the user-specified set of neighbors

$\mathcal{N}(x) \subset \Omega$ described in Section 2. We also present a second version of the algorithm that guarantees stronger results.

The paper is structured as follows. In the next section, we present a definition of local optimality for mixed integer programming, and introduce the relaxed optimality conditions guaranteed by our algorithm. Then in Section 3, we formally describe a general framework for mixed integer pattern search algorithms. In Section 4, we show the existence of a subsequence of iterates converging to an accumulation point that satisfies the relaxed optimality conditions. The key in obtaining this accumulation point lies in considering the unsuccessful iterations, i.e., the iterations where no trial point yielding decrease in the objective function were obtained. A stronger version of the algorithm is also presented. It requires more function evaluations per iteration but guarantees an additional necessary optimality condition. The difference between the two versions of the algorithm is illustrated on a small example in the last subsection.

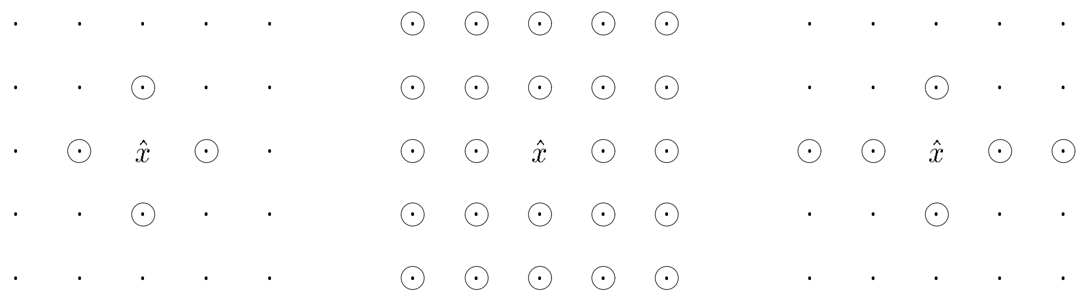
2 Local optimality for mixed integer variables

In the absence of integer variables, the definition of local optimality is straightforward: $\hat{x}$ is a local minimizer of the function f if there exists an $\epsilon > 0$ such that $f(\hat{x}) \leq f(y)$ for all y in a ball $B(\epsilon, \hat{x})$ of radius ϵ around $\hat{x}$.

When the optimization problem contains only discrete variables, a definition of local optimality might be: $f(\hat{x}) \leq f(y)$ for all y in $\mathcal{N}(\hat{x})$, where $\mathcal{N}(\hat{x})$ is a finite set of neighbors around the discrete variable $\hat{x}$. It determines the quality of the solution that one desires from the algorithm, thus defining the notion of "local optimality" one wishes to achieve with respect to the discrete variables. For example, the set $\mathcal{N}(\hat{x})$ might be equal to $\{y \in \mathrm{Z}^{n_d} : \|y - \hat{x}\|_1 \leq 1\}$. We assume that $\mathcal{N}(\hat{x})$ contains $\hat{x}$.

Figure 1 illustrates three types of sets of neighbors by displaying the points of $\mathcal{N}(\hat{x})$ (except $\hat{x}$) as circled dots. Under each figure is the formal definition of the set $\mathcal{N}(\hat{x})$. The first one defines the set of neighbors as the points whose distance from x_k is within one using the one-norm. The second allows a distance of two with the infinity norm. The third one uses a weighted norm. The definition of a neighborhood is flexible enough so that any finite set of integer points can be used to define $\mathcal{N}$.

The set of neighbors need not be represented through the translation of $\mathcal{N}(0)$ as above. For example the Quadratic Assignment Problem in which n facilities must be assigned to n locations: each assignment may be represented using one of the $n!$ permutations of the vector $(1, 2, \dots, n)$. The set of neighbors of a given assignment x could be for example all permutations that differ from x in at most two locations. Consider the instance with three facilities. It may be modeled with three discrete variables ($x^d \in \mathrm{Z}^3$). Not all the points of the



$\{y : \|y - \hat{x}\|_1 \leq 1\}$ $\{y : \|y - \hat{x}\|_\infty \leq 2\}$ $\{y : |y_1 - \hat{x}_1| + 2|y_2 - \hat{x}_2| \leq 2\}$

One-norm Infinity norm Weighted norm

Figure 1: Examples of sets of neighbors $\mathcal{N}(\hat{x})$ of $\hat{x}$.

integer lattice Z^3 represent feasible assignments, only the permutations of $(1, 2, 3)$ are. Also, the ordering is not the classical one associated with a distance: with the set of neighbors $\mathcal{N}(1, 2, 3) = \{(1, 2, 3), (1, 3, 2), (3, 2, 1), (2, 1, 3)\}$ the assignment $(3, 2, 1)$ is nearer to $(1, 2, 3)$ than $(3, 1, 2)$ is.

For mixed integer programming, the definition of local optimality must take into account variations of both the continuous and discrete variables. We propose the following definition: a solution $\hat{x} = (\hat{x}^c, \hat{x}^d) \in \mathrm{R}^{n^c} \times \mathrm{Z}^{n^d}$ is said to be a local minimizer of f if there exists an $\epsilon > 0$ such that

$$f(\hat{x}) \leq f(z) \quad \text{for any} \quad z \in \bigcup_{y \in \mathcal{N}(\hat{x})} \left(B(\epsilon, y^c) \times y^d\right)$$

(where $\mathcal{N}(\hat{x})$ is a finite set of points defined by the user as above). We require a notion of continuity concerning the set of neighbors: if (x_k) is a sequence that converges to $\hat{x}$ then $\mathcal{N}(x_k)$ converges to $\mathcal{N}(\hat{x})$, i.e., for any $\epsilon > 0$ and $\hat{y} \in \mathcal{N}(\hat{x})$, there exists an $y_k \in \mathcal{N}(x_k)$ such that $y_k \in B(\epsilon, y)$.

This definition guarantees that there are no better solutions than $\hat{x}$ in any of the balls (in the continuous space) around the points in the user-defined set of neighbors.

Observe that when there are no discrete variables, or else no continuous ones, this definition reduces to the appropriate one presented above.

The pattern search algorithm for continuous variables only presented in [11] does not necessarily converge to a local minimizer of the function (see Audet [1] for illustrative examples). It guarantees convergence to a relaxation of this optimality condition, namely condition (1). Our algorithm relaxes the local optimality condition for mixed integer programming. The limit point $\hat{x}$ produced by it is such that

$$\nabla^c f(\hat{x}) = 0 \tag{2}$$

(where $\nabla^c f(x) \in \mathrm{R}^{n^c}$ denotes the gradient of f with respect to the continuous variables x^c

while keeping the discrete ones x^d fixed) and for any $\hat{y}$ in the set of neighbors $\mathcal{N}(\hat{x})$

$$f(\hat{x}) \quad \leq \quad f(\hat{y}). \tag{3}$$

In the cases where $f(\hat{y}) < f(\hat{x}) + \xi$ (for a specified $\xi > 0$) then there exists a point $\hat{z}$ whose discrete components $\hat{z}^d$ are identical to $\hat{y}^d$ that satisfies $f(\hat{x}) \leq f(\hat{z}) \leq f(\hat{y})$ and

$$\nabla^c f(\hat{z}) \quad = \quad 0. \tag{4}$$

Furthermore, in the cases where $f(\hat{x}) = f(\hat{y})$ and $\hat{y} \neq \hat{z}$ then

$$f(\hat{x}) \quad = \quad f(\hat{\gamma}). \tag{5}$$

for an infinite number of intermediate points $\hat{\gamma}$ between $\hat{y}$ and $\hat{z}$ (we show in Section 4.2 how to construct these intermediate points). Moreover, we present a stronger version of the algorithm that guarantees that

$$\nabla^c f(\hat{\gamma}) \quad = \quad 0 \tag{6}$$

whenever $f(\hat{x}) = f(\hat{y})$.

3 Pattern search methods

The underlying structure of a pattern search algorithm is as follows. It is an iterative method that generates a sequence of feasible iterates whose objective function value is non-increasing. At any given iteration, the objective function is evaluated at a finite number of points on a mesh in order to find one that yields a decrease in the objective function value.

Any iteration k of a pattern search method is initiated with the *incumbent solution* x_k, i.e., the currently best found solution, as well as with an enumerable subset M_k of the domain $\Omega = \mathrm{R}^{n^c} \times \mathrm{Z}^{n^d}$. Construction of the mesh M_k is formally described in Section 3.1. The objective pursued during each iteration is to obtain a solution on a subset of the current mesh whose function value is strictly less than the incumbent value.

Exploration of the mesh is conducted in one or two phases. First, a finite search, free of any other rules imposed by the algorithm, is performed anywhere on the mesh. Any strategy can be used, as long as it searches finitely many points (including none). If the first search does not succeed in improving the incumbent, the second phase is called. A potentially exhaustive (but always finite) search in small neighborhoods around x_k and around the points in its set of neighbors intersected with the mesh is performed. The set of points visited by this second search is referred to as the *poll set*. Rules for constructing the poll set are detailed in Section 3.2. The first phase (called the *Search step*) provides flexibility to the method and determines in practice the global quality of the solution. The second phase (called the *Poll step*) follows stricter rules and guarantees theoretical convergence.

If a solution having an objective function value less than the incumbent is found in either phase, then the iteration is declared *successful*. The incumbent solution is then updated, and the next iterate is initiated with a (possibly) coarser mesh around the newly found incumbent solution.

Otherwise, the iteration is declared *unsuccessful*. The next iteration is initiated at the same incumbent solution, but with a finer mesh on the continuous variables, and a set of neighbors closer to the incumbent solution. A key property of the mesh exploration is that if an iteration is unsuccessful, then the current objective function value is less than or equal to the objective function values evaluated at all points in the poll set.

In order to properly present the pattern search algorithm, we first detail in the following subsections the construction of the mesh and the poll set.

3.1 The mesh

At any given iteration k, the *current mesh* M_k is a discrete set of points in Ω from which the algorithm selects the next iterate. The coarseness or fineness of the mesh is dictated by the strictly positive *mesh size parameter* $\Delta_k \in \mathrm{R}_+$. Both the mesh and mesh size parameter are updated at every iteration.

The mesh is the direct product of the finite union of lattices in R^{n^c} with the integer space Z^{n^d}. The presentation of the lattices differs from that of Torczon [11], but the sets produced are equivalent. Consider the *basis matrix* $\beta \in \mathrm{R}^{n^c \times n^c}$ and for ℓ varying from 1 to a finite number ℓ_{max}, consider the *generating matrices* $C_\ell \in \mathrm{Z}^{n^c \times n^c}$, then define the *pattern matrices* $P_\ell \in \mathrm{R}^{n^c \times n^c}$ to be the products βC_ℓ. The continuous variables are chosen from one of the translated (by x_k^c) integer lattices

$$L_\ell(\Delta_k) = \left\{ x_k^c + \Delta_k P_\ell z : z \in Z^{n^c} \right\},$$

for $\ell = 1, 2, \ldots, \ell_{max}$. The continuous part x_k^c of the current iterate belongs to each of the ℓ_{max} lattices regardless of the value of the parameter Δ_k. The basis matrix β is constant over all iterations. However in practice, the generating matrices C_ℓ (and thus P_ℓ) that define the lattices can be determined as the algorithm unfolds, as long as only a finite number of them is generated.

Each of these lattices is enumerable, and the minimum distance between two distinct points is proportional to the mesh size parameter Δ_k. When an iteration is successful, the continuous part of the next iterate is chosen in any of these lattices, and thus belongs to their union $M(\Delta_k) = \bigcup_{\ell=1}^{\ell_{max}} L_\ell(\Delta_k)$, the discrete part is chosen in the integer lattice Z^{n^d}.

At iteration k, the current mesh is defined to be the direct product

$$M_k = M(\Delta_k) \times \mathrm{Z}^{n^d}.$$

The mesh is completely defined by the current iterate x_k and the mesh size parameter Δ_k. Whether the iteration is successful or not, the next iterate x_{k+1} is selected in the mesh M_k.

In the case where the Search step in the current mesh is unsuccessful, a second exploration phase must be conducted by the algorithm in the poll set. The Poll step verifies if the incumbent solution is a local minimizer on a current set of neighbors to be defined in the next subsection.

3.2 The poll set

Polling occurs when the search step was unable to obtain a point on the current mesh that decreased the incumbent value. Polling is conducted in up to three stages (not necessarily in this order):
- polling with respect to the continuous variables
- polling on the current set of neighbors
- extended polling (in the case where the function value evaluated at a point in the set of neighbors is close to the incumbent value).

Polling with respect to the continuous variables requires the use of positive bases on R^{n^c}. A positive basis is a set of non-zero vectors in R^{n^c} whose non-negative linear combinations span R^{n^c}, but no proper subset does so. Each positive bases contain at least n^c+1 and at most $2n^c$ vectors. These are referred to as minimal and maximal positive bases. The following key property of positive bases is used in this document (see Davis [5] for characterization of positive bases). For any non-zero vector a in R^{n^c} and positive basis B on R^{n^c}, there exists a vector b of the basis B such that

$$a^t b \;<\; 0. \tag{7}$$

Let $\mathcal{B}$ be a finite set of positive bases on R^{n^c} such that every column b of any positive basis of $\mathcal{B}$ is of the form $P_\ell z$ for some $z \in Z^{n^c}$ and $1 \le \ell \le \ell_{max}$. The P_ℓ's are the same matrices used to construct the lattices L_ℓ in Section 3.1. The set $\mathcal{B}$ is fixed throughout all iterations. The polling points of this first step are obtained by scaling a basis B of $\mathcal{B}$ by the mesh size parameter as follows: at iteration k, define $\mathcal{N}^c(x)$, the *mesh neighborhood of the continuous variables around* x, to be

$$\mathcal{N}^c(x) \;=\; \{x + \Delta_k (b,0)^t : b \in B_k(x)\} \tag{8}$$

for some positive basis $B_k(x) \in \mathcal{B}$ that depends on both the iteration number k and the point x. This definition ensures that the mesh neighborhood $\mathcal{N}^c(x_k)$ is a subset of the current mesh

M_k. Moreover, $\mathcal{N}^c(x)$ is constructed using a single positive basis chosen from a finite set, and thus there are only a finite number of such neighborhoods to choose from.

The motivation for introducing positive bases for the continuous variables is that if the gradient $\nabla^c f$ of the function f with respect to the continuous variables is non-zero, then at least one of the basis vectors defines a descent direction. The original work of Torczon [11] uses a maximal positive basis. It was latter generalized in Lewis and Torczon [9] to any positive basis, thus reducing the minimum number of points in the polling set from $2n^c$ to $n^c + 1$.

The second stage of the polling step depends on the set of neighbors $\mathcal{N}$ defined by the user. In order to allow varying the definition of the set of neighbors for a finite number of iterations, we define the *current set of neighbors* $\mathcal{N}_k = \{y_k^1, y_k^2, \dots, y_k^{i_k}\} \subset M_k$ where i_k is a finite integer to be such that $\mathcal{N}_k$ differs from $\mathcal{N}(x_k)$ at most at a finite number of iterations k. This flexibility allows finitely many redefinitions of $\mathcal{N}_k$ to adjust the cost of a Poll step (see Section 3.3). For example, if the user defines $\mathcal{N}$ through the infinity norm (as in the second example of Figure 1), it might be worthwhile in the first few iterations to define $\mathcal{N}_k$ through the one-norm (as in the first example of Figure 1). Then, once a solution satisfying the one-norm is obtained, the set of neighbors $\mathcal{N}_k$ may be changed by using the infinity norm.

If none of the above-mentioned polling points (i.e., those in $\mathcal{N}^c(x_k)$ and in $\mathcal{N}_k$) yield decrease in the objective function value, an extended polling step might be required before declaring the iteration unsuccessful. Let $\xi > 0$ be a given function value tolerance (usually provided by the user). An extended poll must be conducted around each point of the set of neighbors $\mathcal{N}_k$ of x_k at which the function value is within ξ of $f(x_k)$. Intuitively, ξ represents a tolerance which is such that if a discrete neighbor y_k^i in $\mathcal{N}_k$ provides such a near function value, then slightly changing the continuous components of y_k^i by extended polling may produce a new best solution.

More precisely, consider any point y_k^i in the set of neighbors $\mathcal{N}_k$. In the case where $f(y_k^i) > f(x_k) + \xi$ or $f(y_k^i) \le f(z)$ for all z in $\mathcal{N}^c(y_k^i)$, set $j_k^i = 0$: this means that the poll step need not be extended. In all other cases, set $y_{k,0}^i = y_k^i$ and for $j = 1, 2, \dots$ iteratively choose $y_{k,j}^i$ in the mesh neighborhood $\mathcal{N}^c(y_{k,j-1}^i)$ in such a way that $f(y_{k,j}^i) < f(y_{k,j-1}^i)$ until it is no longer possible. The last point (whose index is denoted $j = j_k^i$) satisfies $f(y_{k,j_k^i}^i) < f(z)$ for all z in $\mathcal{N}^c(y_{k,j_k^i}^i)$. Define z_k^i to be the endpoint $y_{k,j_k^i}^i$ of the extended poll step.

With this construction, the function values $f(y_k^i) = f(y_{k,0}^i), f(y_{k,1}^i), \dots, f(y_{k,j_k^i}^i) = f(z_k^i)$ are monotonically decreasing. Only at the last point z_k^i is the function required to be evaluated at every point of its mesh neighborhood $\mathcal{N}^c(z_k^i)$. Observe that j_k^i may be 0, in which case $y_k^i = z_k^i$. This happens either when $f(y_k^i) > f(x_k) + \xi$ or $f(y_k^i) \le f(z)$ for all z in $\mathcal{N}^c(y_k^i)$. The index j_k^i are all finite since for given k and i, all points $y_{k,j}^i$ belong to the mesh M_k intersected with the compact level set $L(x_0)$.

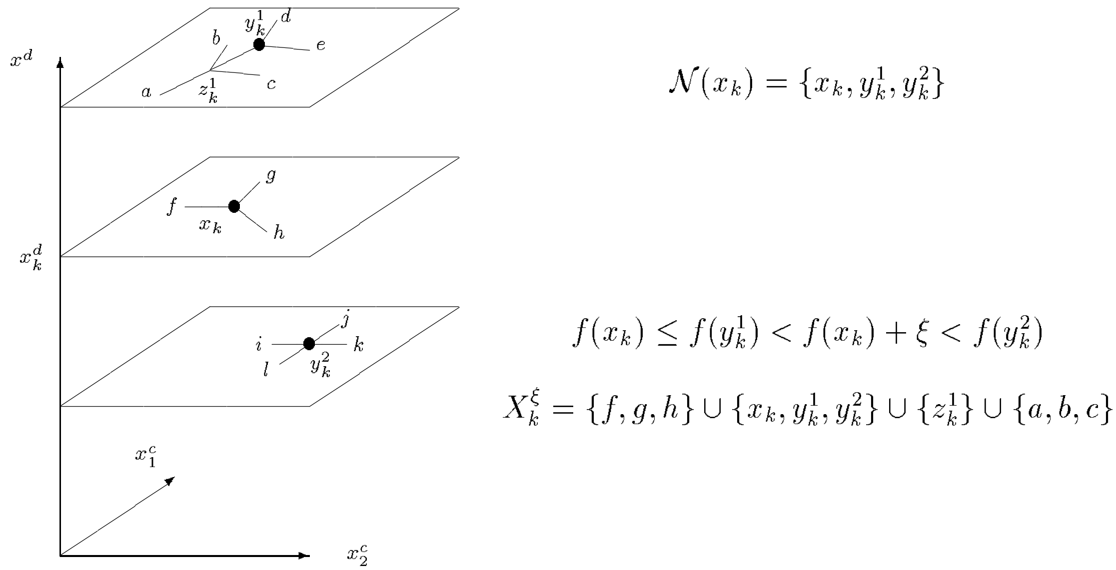


Figure 2: Construction of the current mesh neighborhood X^ξ_k.

The set of all polling points at iteration k is denoted X^ξ_k and may be written explicitly as

$$X^\xi_k = \mathcal{N}^c(x_k) \bigcup \mathcal{N}_k \bigcup \left\{ y^i_{k,j} : \begin{array}{c} i = 1, 2, \ldots, i_k, \\ f(y^i_k) \le f(x_k) + \xi, \\ j = 1, 2, \ldots, j^i_k \end{array} \right\} \bigcup \left\{ z : \begin{array}{c} i = 1, 2, \ldots, i_k, \\ f(y^i_k) \le f(x_k) + \xi, \\ z \in \mathcal{N}^c(y^i_{k,j^i_k}) \end{array} \right\},$$

or as the equivalent set

$$X^\xi_k = \mathcal{N}^c(x_k) \bigcup \mathcal{N}_k \bigcup_{\substack{i=1,2,\ldots,i_k, \\ f(y^i_k) \le f(x_k)+\xi}} \{y^i_{k,j} : j = 1, 2, \ldots, j^i_k\} \cup \mathcal{N}^c(y^i_{k,j^i_k}),$$

where the points $y^i_{k,j}$ are determined by the procedure detailed above.

Figure 2 illustrates an instance in which there are two continuous variables and one discrete variable. The set of neighbors of the iterate x_k is assumed to be $\mathcal{N}_k = \mathcal{N}(x_k) = \{x_k, y^1_k, y^2_k\}$. Therefore, x_k is a local minimizer of the function f if $f(x_k)$ is less than or equal to the function value evaluated at all points in balls around x_k, y^1_k and y^2_k. The letters a to l in the figure represent mesh neighborhoods of the continuous variables:

$$\mathcal{N}^c(y^1_k) = \{d, e, z^1_k\},\ \mathcal{N}^c(z^1_k) = \{a, b, c\},\ \mathcal{N}^c(x_k) = \{f, g, h\} \text{ and } \mathcal{N}^c(y^2_k) = \{i, j, k, l\}.$$

In this example, since $f(x_k) \le f(y^1_k) < f(x_k) + \xi < f(y^2_k)$ the poll set X^ξ_k contains points in $\mathcal{N}^c(x_k)$ and $\mathcal{N}^c(y^1_k)$ (among others). Assuming that $f(z^1_k) < f(y^1_k)$ but $f(a) \ge f(z^1_k)$, $f(b) \ge f(z^1_k)$ and $f(c) \ge f(z^1_k)$ leads to the poll set $X^\xi_k = \{f, g, h\} \cup \{x_k, y^1_k, y^2_k\} \cup \{z^1_k\} \cup \{a, b, c\}$.

Using the above notation, we can now present the generalized mixed integer pattern search algorithm.

3.3 The generalized mixed integer pattern search algorithm

Our presentation of the pattern search algorithm is closer to that of Booker *et al.* [3] than to that of Torczon [11]. Consider the given the initial mesh $M_0 \subset \Omega$ with mesh size parameter Δ_0 and initial point x_0 of M_0. Also, let $\xi > 0$ be the objective function change tolerance used to trigger extended polling in the construction of the poll set. Recall that if $f(y_k) \le f(x_k) + \xi$ for some y_k in the set of neighbors $\mathcal{N}_k$ then the polling step must be extended around y_k.

Throughout the document, the following assumptions are made:

(A1) The level set $L(x_0) = \{x \in \Omega : f(x) \le f(x_0)\}$ is compact.

(A2) f is continuously differentiable over a neighborhood of $L(x_0)$ when variables in Z^{n^d} are fixed, i.e., for any $x^d \in \mathrm{Z}$ the function $f_{x^d} : \mathrm{R}^{n^c} \to \mathrm{R}$ where $x^c \mapsto f(x^c, x^d)$ is continuously differentiable over a neighborhood of $\{x^c : (x^c, x^d) \in L(x_0)\}$.

At any iteration $k \ge 0$, the general rules for choosing x_{k+1} in the current mesh M_k and obtaining the next mesh size parameter Δ_{k+1} are as follows.

Algorithm GMIPS: General Mixed-Integer Pattern Search

1. Search step (in current mesh). Employ some finite strategy to obtain an $x_{k+1} \in M_k$ satisfying $f(x_{k+1}) < f(x_k)$. If such an x_{k+1} is found, declare the Search step (as well as the iteration) successful, then expand the mesh at Step 3.

2. Poll step. This step is reached only if the Search step is unsuccessful. If $f(x_k) \le f(x)$ for every x in the poll set X_k^ξ, then declare the Poll step (as well as the iteration) unsuccessful and shrink the mesh at Step 4. Otherwise, choose $x_{k+1} \in X_k^\xi$ to be a point such that $f(x_{k+1}) < f(x_k)$, declare the Poll step (as well as the iteration) successful, and expand the mesh at Step 3.

3. Mesh expansion (at successful iterations). Let $\Delta_{k+1} = \tau^{m_k^+} \Delta_k$ (for $\tau^{m_k^+} \ge 1$ defined below). Increase k, and initiate the next iteration at Step 1.

4. Mesh reduction (at unsuccessful iterations). Set x_{k+1} to x_k and let $\Delta_{k+1} = \tau^{m_k^-} \Delta_k$ (for $0 < \tau^{m_k^-} < 1$ defined below). Increase k, and initiate the next iteration at Step 1. ∎

In the Search and Poll steps, the number of candidate points among which the next iterate can be chosen is finite, since it must belong to the intersection of the enumerable current mesh and the compact set $L(x_0)$.

The parameters in the two last steps are the rational number $\tau > 1$ and the integers (whose absolute values are bounded above by $m_{max} \ge 0$) $m_k^+ \ge 0$ and $m_k^- \le -1$. In [11], the mesh reduction parameter m_k^- was fixed for all $k \ge 0$. This restriction is relaxed here without affecting the convergence results. We plan to exploit this flexibility in subsequent work to increase the convergence speed.

The conditions on these parameters imply the *simple decrease property* used throughout the document: Iteration k is successful if and only if $f(x_{k+1}) < f(x_k)$, if and only if $\Delta_{k+1} \ge$

Δ_k and, if and only if $x_{k+1} \neq x_k$. Another important implication of the parameters' definition is that if the iteration k is unsuccessful, then $f(x_k) \leq f(x)$ for all $x \in X_k^\xi$ and thus $f(x_k) \leq f(z)$ for all $z \in \mathcal{N}^c(x_k)$ and whenever $f(y_k^i) \leq f(x_k) + \xi$ for some $y_k^i \in \mathcal{N}_k$ then $f(z_k^i) \leq f(z)$ for all $z \in \mathcal{N}^c(z_k^i)$ for all $i = 1, 2, \ldots, i_k$. Moreover, Δ_{k+1} is obtained by multiplying Δ_k by a finite positive or negative integer power of τ. Therefore, for any $k \geq 0$, we can write

$$\Delta_k = \Delta_0 \tau^{r_k}, \tag{9}$$

for some r_k belonging to Z.

Notice that the cost of the Poll step is expected to depend not only on ξ, but also on the definition of the set of neighbors $\mathcal{N}$. Thus, the user can pay more function evaluations for a stronger local integer solution by defining $\mathcal{N}$ to be a larger neighborhood.

4 Proof of convergence

This section contains the convergence proof for the general mixed integer pattern search algorithm. We start by studying the behavior of the mesh size parameter Δ_k. The first important result is that $\liminf_{k \to +\infty} \Delta_k = 0$. Therefore, there is a subsequence of mesh size parameters that converges to zero. It follows that there is an infinite number of unsuccessful iterations.

Second, we analyze a converging subsequence of unsuccessful iterates whose mesh size parameters converge to zero. We show that any accumulation point of the subsequence satisfies the optimality conditions (2)-(5). By focusing on unsuccessful iterations, the result for the continuous variables is shown using a much shorter proof than in [11].

We also present a stronger version of the algorithm that yields a stronger result, i.e., the optimality condition (6). Finally, a small example illustrates in the last subsection the differences between the two versions of the algorithm.

4.1 Boundedness of the mesh size parameters

We prove here that there is a subsequence of mesh size parameters Δ_k that converges to zero. In order to do so, we first show that these parameters are bounded above by a constant, independent of the iteration number k.

Lemma 4.1 *There exists a positive integer r_{UB} such that $\Delta_k \leq \Delta_0 \tau^{r_{UB}}$ for any $k \geq 0$.*

Proof: Let Δ be a mesh size parameter large enough so that the union of lattices $M(\Delta)$ intersects the compact level set $\{x^c : x \in L(x_0)\}$ only at the translation parameter x_k^c, i.e., for any $1 \leq \ell \leq \ell_{max}$ and nonzero $z \in Z^{n^c}$, if x is in $L(x_0)$ then the solution $x^c + \Delta P_\ell z$ does not belong to the projection of $L(x_0)$ on the continuous variables space. Therefore, if at iteration k the mesh size parameter Δ_k is greater than or equal to Δ then

$$M_k \cap L(x_0) \quad \subset \quad \{x_k^c\} \times Z^{n_d}.$$

Moreover, only a finite number of iterations will follow before the mesh size parameter drops below Δ. Indeed, the continuous part of all these iterates will necessarily be equal to x_k^c, and the discrete part of these iterates can only take a finite number of values because $L(x_0)$ is compact. Let d_{max} be the total number of distinct values that the discrete variables may take in the compact set $L(x_0)$. Therefore, there will be no more than d_{max} successful iterations before the mesh size parameter goes below Δ.

Recall that the expansion mesh size control parameter is bounded above by $\tau^{m_{max}}$. Let r_{UB} be a large enough integer so that $\Delta_0 \tau^{r_{UB}} \geq \Delta (\tau^{m_{max}})^{d_{max}}$. It follows that the mesh size parameter at any iteration will never exceed $\Delta_0 \tau^{r_{UB}}$. ∎

We now study the convergence behavior of the mesh size parameter. The proof of this result is essentially identical to that of Torczon [11].

Theorem 4.2 *The mesh size parameters satisfy* $\liminf\limits_{k \to +\infty} \Delta_k = 0$.

Proof: Suppose by contradiction that there exists a negative integer r_{LB} such that $0 < \Delta_0 \tau^{r_{LB}} \leq \Delta_k$ for all $k \geq 0$. Equation (9) states that for every $k \geq 0$ there is $r_k \in Z$ such that $\Delta_k = \tau^{r_k} \Delta_0$. Combining this with Lemma 4.1 implies that for any $k \geq 0$, r_k takes its value among the integers of the bounded interval $[r_{LB}, r_{UB}]$. Therefore, r_k and Δ_k can only take a finite number of values for all $k \geq 0$.

For any k, the continuous part of the next iterate x_{k+1}^c belongs to a lattice P_{ℓ_k} where $1 \leq \ell_k \leq \ell_{max}$, therefore it can be written $x_k^c + \Delta_k P_{\ell_k} z_k$ for some $z_k \in Z^{n^c}$. By substituting $\Delta_k = \Delta_0 \tau^{r_k}$ and $P_\ell = \beta C_{\ell_k}$, it follows that for any integer N

$$x_N^c \;=\; x_0^c + \sum_{k=1}^{N-1} \Delta_k P_{\ell_k} z_k \;=\; x_0^c + \Delta_0 \beta \sum_{k=1}^{N-1} \tau^{r_k} C_{\ell_k} z_k \;=\; x_0^c + \frac{p^{r_{LB}}}{q^{r_{UB}}} \Delta_0 \beta \sum_{k=1}^{N-1} p^{r_k - r_{LB}} q^{r_{UB} - r_k} C_{\ell_k} z_k$$

where p and q are relatively prime integers satisfying $\tau = \frac{p}{q}$.

Since for any k the term $p^{r_k - r_{LB}} q^{r_{UB} - r_k} C_{\ell_k} z_k$ appearing in this last sum is an integer, it follows that the continuous part of all iterates lies on the translated integer lattice generated by x_0^c and the columns of $\frac{p^{r_{LB}}}{q^{r_{UB}}} \Delta_0 \beta$. Moreover, the discrete part of all iterates also lies on the integer lattice Z^{n^d}.

Therefore, since all iterates belong to the compact set $L(x_0)$, it follows that there is only a finite number of different iterates, and thus one of them must be visited infinitely many times. Simple decrease ensures that the mesh size parameters converge to zero, which is a contradiction. ∎

4.2 The main results

Torczon [11] shows that condition (1) holds, i.e., there exists an accumulation point $\hat{x}$ of the sequence of iterates such that $\nabla^c f(\hat{x}) = 0$. Through a shorter proof, we show a stronger result. We show the existence of an accumulation point $\hat{x}$ of the sequence of *unsuccessful* iterates that satisfies (2) and is a local optimizer with respect to the set of neighbors $\mathcal{N}(\hat{x})$ in the sense of conditions (3), (4) and (5). Recall that iteration k is unsuccessful if and only if $x_{k+1} = x_k$, which is equivalent to $\Delta_{k+1} < \Delta_k$. Thus, the number of unsuccessful iterations is infinite since $\liminf_{k\to+\infty} \Delta_k = 0$.

Consider the indices of the unsuccessful iterations whose corresponding mesh size parameters go to zero. For any accumulation point of such a sequence, there is an iterate x_k arbitrarily close to it for which no polling point of the set X_k^ξ yields descent. The following proposition details properties of an accumulation point $\hat{x}$ of the sequence of unsuccessful iterations whose mesh size parameters converge to 0. The notation $y^i = (y^{ic}, y^{id})$ is employed to partition the variable y^i into its continuous $(y^i)^c$ and discrete $(y^i)^d$ components.

Proposition 4.3 *There is a point $\hat{x} \in L(x_0)$ and a subset of indices of unsuccessful iterates $K \subset \{k : x_{k+1} = x_k\}$ such that*

$$\lim_{k\in K} \Delta_k = 0, \qquad \lim_{k\in K} x_k = \hat{x} \quad \textit{and} \quad \mathcal{N}_k = \mathcal{N}(x_k)\ \forall k \in K.$$

Moreover, if $\{\hat{y}^1, \hat{y}^2, \ldots, \hat{y}^{\hat{\imath}}\}$ denotes the $\hat{\imath}$ neighbors in the set $\mathcal{N}(\hat{x})$, then for each $i \in \{1, 2, \ldots, \hat{\imath}\}$ there exists a $\hat{z}^i = (\hat{z}^{ic}, \hat{y}^{id})$ such that

$$\lim_{k\in K} y_k^i = \hat{y}^i \quad \textit{and} \quad \lim_{k\in K} z_k^i = \hat{z}^i,$$

where the z_k^i are the endpoints of the extended poll steps.

Proof: Theorem 4.2 guarantees that $\liminf_{k\to+\infty} \Delta_k = 0$, thus there is an infinite subset of indices of unsuccessful iterations $K' \subset \{k : x_{k+1} = x_k\} = \{k : \Delta_{k+1} < \Delta_k\}$ such that the subsequence $(\Delta_k)_{k\in K'}$ converges to zero.

Since all iterates x_k lie in the compact set $L(x_0)$, we can extract an infinite subset $K'' \subset K'$ such that the subsequence $(x_k)_{k\in K''}$ converges. Let $\hat{x}$ in $L(x_0)$ be the limit point of such a subsequence.

Moreover, since $\mathcal{N}_k$ differs from $\mathcal{N}(x_k)$ at most at a finite number of iterates, we may assume without any loss of generality that $x_k^d = \hat{x}^d$ for all $x_k \in K''$.

Let $\hat{y}^i \in \mathcal{N}(\hat{x})$ be an accumulation point of the sequence $y_k^i \in \mathcal{N}_k$, and let $\hat{z}^i$ be an accumulation point of the sequence z_k^i of endpoints of the extended poll step initiated at

y_k^i. Recall that the endpoint z_k^i is equal to y_k^i in the case that the extended poll step is not required.

Choose $K \subset K''$ to be such that $\lim_{k \in K} y_k^i = \hat{y}^i$ and $\lim_{k \in K} z_k^i = \hat{z}^i$ for each $i \in \{1, 2, \dots, \hat{\imath}\}$. ∎

Torczon [11] observes that setting the mesh size increase parameter m_{max} to zero (in the mesh expansion step of the GPS algorithm) ensures that $\lim_{k \to \infty} \Delta_k = 0$. The same holds for our GMIPS algorithm. It follows that in this case, all the convergence results below hold for every accumulation point of the sequence of unsuccessful iterates.

For the rest of this subsection, we assume that $\hat{x}$ and K satisfy the conditions of Proposition 4.3. The main results can now be proved. We first show that $\hat{x}$ is a local optimal solution with respect to the set of neighbors $\mathcal{N}(\hat{x})$.

Theorem 4.4 *The accumulation point $\hat{x}$ satisfies $f(\hat{x}) \leq f(\hat{y}^i)$ for all $\hat{y}^i \in \mathcal{N}(\hat{x})$.*

Proof: Suppose by contradiction that there is a $\hat{y}^i \in \mathcal{N}(\hat{x})$ such that $f(\hat{x}) > f(\hat{y}^i)$.

Continuity of the function f with respect to the continuous variables guarantees the existence of an $\epsilon > 0$ such that if z belongs to the ball $B(\epsilon, \hat{y}^i)$ centered at $\hat{y}^i$ of radius ϵ then $f(z) < f(\hat{x})$.

Proposition 4.3 guarantees that the subsequences $(x_k)_{k \in K}$ and $(y_k^i)_{k \in K}$ respectively converge to $\hat{x}$ and $\hat{y}^i$. We required in Section 2 that the set $\mathcal{N}(x_k)$ converges for $k \in K$ to $\mathcal{N}(\hat{x})$ in the sense that if $k \in K$ is large enough, then there exists a $y_k^i \in \mathcal{N}(x_k)$ such that $y_k^i \in B(\epsilon, \hat{y}^i)$.

Therefore, there exists an iteration $k \in K$ such that y_k^i belongs to $\mathcal{N}_k \cap B(\epsilon, \hat{y}^i)$ and satisfies $f(y_k^i) < f(\hat{x}) \leq f(x_k)$. It follows that the iteration is successful, contradicting the fact that k belongs to $K \subset \{k : x_{k+1} = x_k\}$. ∎

In the case where the inequality in Theorem 4.4 is strict, i.e., $f(\hat{x}) < f(\hat{y}^i)$, then the notion of local optimality for mixed integer programming presented in Section 2 is verified: there exists a $\epsilon > 0$ such that $f(\hat{x}) \leq f(z)$ for any z in a ball of radius ϵ around $\hat{y}^i$. This follows from the continuity of the function f.

Next, we study the gradient of the function f with respect to the continuous variables at the accumulation point $\hat{x}$. The proof of Theorem 4.5 for the continuous case is much shorter than the original one of Torczon [11].

Theorem 4.5 *The accumulation point $\hat{x}$ satisfies $\nabla^c f(\hat{x}) = 0$.*

Proof: Equation (8) and the mean value theorem imply that any unsuccessful iteration k

satisfies

$$\begin{aligned} f(x_k) \ \leq \ \min_{z\in\mathcal{N}^c(x_k)} f(z) &= \min_{z\in\{x_k+\Delta_k(b,0)^t : b\in B_k(x_k)\}} f(z) \\ &= \min_{b\in B_k(x_k)} f(x_k+\Delta_k(b,0)^t) \\ &= \min_{b\in B_k(x_k)} f(x_k)+\Delta_k\nabla^c f(x_k+\alpha_{b,k}\Delta_k(b,0)^t)^t b \\ &= f(x_k)+\Delta_k \min_{b\in B_k(x_k)} \nabla^c f(x_k+\alpha_{b,k}\Delta_k(b,0)^t)^t b \end{aligned}$$

for some $\alpha_{b,k} \in [0,1]$ that depends on b and k. Therefore

$$0 \ \leq \ \min_{b\in B_k(x_k)} \nabla^c f(x_k+\alpha_{b,k}\Delta_k(b,0)^t)^t b.$$

Taking the limit for $k \in K$ yields $0 \leq \min\limits_{b\in B} \nabla^c f(\hat{x})^t b$ (by Proposition 4.3) for at least one positive basis B of the finite set $\mathcal{B}$ since the function f is assumed to be continuously differentiable. The positive basis property (7) guarantees that $\nabla^c f(\hat{x}) = 0$. ∎

Audet [1] shows through a small example containing only continuous variables that this result cannot be strengthened to $\lim_{k\to\infty} \|\nabla^c f(x_k)\| = 0$ since there may be an accumulation point whose gradient is non-zero. It is also shown there that no second-order optimality conditions can be guaranteed, which is not surprising for an algorithm that uses only function values.

The following result shows that the gradient norm at the end points of the extended poll converges to zero for $k \in K$. The proof is similar to that of Theorem 4.5.

Theorem 4.6 *The accumulation point $\hat{x}$ and any $\hat{y}^i \in \mathcal{N}(\hat{x})$ satisfying $f(\hat{y}^i) < f(\hat{x})+\xi$ are such that $\nabla^c f(\hat{z}^i) = 0$ where $\hat{z}^i$ is any limit point of the extended poll endpoints.*

Proof: The result is shown in Theorem 4.5 for $\hat{y}^i = \hat{x}$. Let $i \in \{1,2,\ldots,\hat{\imath}\}$ be such that $\hat{y}^i \neq \hat{x}$ and $f(\hat{y}^i) < f(\hat{x})+\xi$. Then by the extended polling algorithm of Section 3.2, any unsuccessful iteration k satisfying $f(y_k^i) \leq f(x_k)+\xi$ is such that

$$\begin{aligned} f(z_k^i) \ \leq \ \min_{z\in\mathcal{N}^c(z_k^i)} f(z) &= \min_{z\in\{z_k^i+\Delta_k(b,0)^t : b\in B_k(z_k^i)\}} f(z) \\ &= \min_{b\in B_k(z_k^i)} f(z_k^i+\Delta_k(b,0)^t) \\ &= \min_{b\in B_k(z_k^i)} f(z_k^i)+\Delta_k\nabla^c f(z_k^i+\alpha_{b,k}^i\Delta_k(b,0)^t)^t b \\ &= f(z_k^i)+\Delta_k \min_{b\in B_k(z_k^i)} \nabla^c f(z_k^i+\alpha_{b,k}^i\Delta_k(b,0)^t)^t b \end{aligned}$$

for some $\alpha_{b,k}^i \in [0,1]$ that depends on i, b and k. Therefore

$$0 \ \leq \ \min_{b\in B_k(z_k^i)} \nabla^c f(z_k^i+\alpha_{b,k}^i\Delta_k(b,0)^t)^t b.$$

Taking the limit for $k \in K$ yields $0 \leq \min_{b \in B} \nabla^c f(\hat{z}^i)^t b$ (by Proposition 4.3) for at least one positive basis B of the finite set $\mathcal{B}$ since the function f is assumed to be continuously differentiable. The positive basis property (7) guarantees that $\nabla^c f(\hat{z}^i) = 0$. ∎

The next result shows that the function is constant at an infinite number of intermediate points between $\hat{y}^i$ and $\hat{z}^i$ whenever $f(\hat{y}^i) = f(\hat{x})$.

Proposition 4.7 *The accumulation point $\hat{x}$ and any $\hat{y}^i \in \mathcal{N}(\hat{x})$ satisfying $f(\hat{y}^i) = f(\hat{x})$, are such that any accumulation point $\hat{\gamma}^i$ of the sequence of extended poll points $(y^i_{k,j})$ satisfies $f(\hat{\gamma}^i) = f(\hat{x})$. Moreover, if $\hat{y}^i \neq \hat{z}^i$, then there are infinitely many of these accumulation points.*

Proof: Let $\hat{y}^i$ in $\mathcal{N}(\hat{x})$ be such that $f(\hat{y}^i) = f(\hat{x})$. Let $\hat{\gamma}^i$ be an accumulation point of the sequence of extended poll points $(y^i_{k,j})$.

Since $f(\hat{x}) \leq f(y^i_{k,j+1}) < f(y^i_{k,j})$ for $j = 0, 1, \ldots, j^i_k - 1$ and since $\left(f(y^i_{k,0})\right)_{k \in K}$ converges to $f(\hat{x})$, then $f(\hat{x}) = f(\hat{\gamma}^i)$.

To show the second part of the result, we first let $d = \|\hat{y}^i - \hat{z}^i\|$ be the non-zero distance between $\hat{y}^i$ and $\hat{z}^i$. Second, for any p in $]0, d[$, we define the set

$$Y_p \;=\; \{y^i_{k,j} : \; k \in K, \; j \in \{0, 1, \ldots, j^i_k - 1\}, \; \|y^i_{k,j} - \hat{y}^i\| \leq p, \; \|y^i_{k,j+1} - \hat{y}^i\| > p\|\}.$$

Since $y^i_{k,0} \to \hat{y}^i$ and $y^i_{k,j^i_k} \to \hat{z}^i$, it follows that the set Y_p contains infinitely many points for any p in $]0, 1[$. Any accumulation point $\hat{\gamma}^i_p$ of Y_p satisfies $\|\hat{\gamma}^i_p - \hat{y}^i\| = p$ since Δ_k converges to 0 (in K) and $y^i_{k,j+1} = y^i_{k,j} + \Delta_k b^i_{k,j}$ for some vector $b^i_{k,j}$ of the basis $B_k(y^i_{k,j})$ of the finite set $\mathcal{B}$. Therefore, if $p \neq q$ then $\hat{\gamma}^i_p \neq \hat{\gamma}^i_q$ and the result follows. ∎

4.3 Stronger results

Theorem 4.6 may be strengthened under the following (more expensive) version of extended polling.

Strong Extended Polling Step:

$$y^i_{k,j+1} \in \arg \min_{y \in \mathcal{N}^c(y^i_{k,j})} f(y) \quad \text{for} \quad i = 1, 2, \ldots, i_k \text{ and } j = 0, 1, \ldots, j^i_k - 1.$$

This requires performing a complete extended poll step, i.e., $y^i_{k,j+1}$ is chosen only after evaluating the function value at all points of the continuous mesh neighborhood around $y^i_{k,j}$ (and retaining the one that yields the smallest value).

The following result bounds the decrease in the objective function value under precise conditions.

Proposition 4.8 *If the accumulation point $\hat{x}$ and the point $\hat{y}^i \in \mathcal{N}(\hat{x})$ were obtained under the Strong Extended Polling Step, then for any $\eta > 0$, there exist $\delta > 0$ independent of the iteration number k, such that all iterates for which $\Delta_k < \delta$, $y_k^{ic} = \hat{y}^{ic}$ and $\min_{b \in B_k(y^i_{k,j})} \nabla^c f(y^i_{k,j})^t b \geq \frac{-\eta}{3}$, for some $j \in \{0, 1, \ldots, j^i_k - 1\}$ also satisfy*

$$f(y^i_{k,j}) - f(y^i_{k,j+1}) \geq \frac{\eta \|y^i_{k,j} - y^i_{k,j+1}\|}{4 \max\{\|b\| : b \in B \in \mathcal{B}\}}.$$

Proof: For any iteration k such that $y_k^{ic} = \hat{y}^{ic}$ define $b^i_{k,j}$ to be a basis vector of the set $\arg \min_{b \in B_k(y^i_{k,j})} \nabla^c f(y^i_{k,j})^t b$. The Strong Extended Polling Step, combined with the mean value theorem guarantee that

$$f(y^i_{k,j+1}) \;\leq\; f(y^i_{k,j} + \Delta_k b^i_{k,j}) \;=\; f(y^i_{k,j}) + \Delta_k \nabla^c f(w^i_{k,j})^t b^i_{k,j}. \tag{10}$$

for some $w^i_{k,j} = y^i_{k,j} + \alpha^i_{k,j} \Delta_k b^i_{k,j}$ where $\alpha^i_{k,j}$ belongs to the interval $[0, 1]$.

Consider the sets parameterized by some scalar $\eta > 0$

$$Y = \left\{ y^i_{k,j} : -\nabla^c f(y^i_{k,j})^t b^i_{k,j} \geq \frac{\eta}{3} \right\} \subset L(x_0)$$

and

$$W = \left\{ w^i_{k,j} : -\nabla^c f(w^i_{k,j})^t b^i_{k,j} \leq \frac{\eta}{4} \right\}$$

We assume that $Y \neq \emptyset$ otherwise the result would be trivial. Theorem 4.6 guarantees that $W \neq \emptyset$ since the subsequence of gradients $\left(\nabla^c f(w^i_{k,j^i_k}) \right)_{k \in K}$ converges to the limit of $(\nabla^c f(z^i_k))_{k \in K}$ which is simply $\nabla^c f(\hat{z}^i) = 0$.

Since the function f is assumed continuously differentiable with respect to the continuous variables over a neighborhood of $L(x_0)$, the distance $dist(Y, W)$ between these two disjoint sets is strictly positive. Let $\delta = \frac{dist(Y,W)}{2 \max\{\|b\| : b \in B \in \mathcal{B}\}}$ be a strictly positive finite number independent of k.

Moreover, if $y^i_{k,j}$ belongs to the set Y and if $\Delta_k < \delta$ then the point $w^i_{k,j}$ does not belong to W $\left(\text{since } \|w^i_{k,j} - y^i_{k,j}\| \leq \Delta_k \|b^i_{k,j}\| < \frac{dist(Y,W)}{2}\right)$ and thus $-\nabla^c f(w^i_{k,j})^t b^i_{k,j} \geq \frac{\eta}{4}$. Combining this with equation (10) yields

$$f(y^i_{k,j+1}) \leq f(y^i_{k,j}) - \frac{\eta \Delta_k}{4} = f(y^i_{k,j}) - \frac{\eta \|y^i_{k,j} - y^i_{k,j+1}\|}{4 \max\{\|b\| : b \in B \in \mathcal{B}\}}$$

which concludes the proof. ∎

The following result strengthens Theorem 4.6 by showing that the gradient norm is zero at the accumulation points $\hat{\gamma}^i$ of Proposition 4.7.

Theorem 4.9 *If the accumulation point $\hat{x}$ and the point $\hat{y}^i \in \mathcal{N}(\hat{x})$ were obtained under the Strong Extended Polling Step, and if $f(\hat{y}^i) = f(\hat{x})$ then $\nabla^c f(\hat{\gamma}^i) = 0$ for each accumulation point $\hat{\gamma}^i$ of the sequence $(y^i_{k,j})$.*

Proof: Suppose by contradiction that there exists an accumulation point $\hat{\gamma}^i$ of the sequence $(y^i_{k,j})$ that satisfies $\|\nabla^c f(\hat{\gamma}^i)\| \neq 0$. Set

$$-\eta \;=\; \max_{B\in\mathcal{B}} \min_{b\in B} \nabla^c f(\hat{\gamma}^i)^t b \;<\; 0. \tag{11}$$

Let δ be the positive parameter derived from Proposition 4.8 that depends only on $\eta < 0$.

Continuous differentiability of f over the compact set $L(x_0)$ allows us to define $\epsilon > 0$ to be such that if u and v in Ω satisfy $\|u - v\| < \epsilon$ then

$$\left|\min_{b\in B} \nabla^c f(u)^t b - \min_{b\in B} \nabla^c f(v)^t b\right| \;<\; \frac{\eta}{3} \tag{12}$$

for every basis B of the finite set $\mathcal{B}$.

Consider a subsequence $(y^i_{k,j(k)})$ (where $j(k) \in \{1, 2, \ldots j^i_k\}$ is increasing) that converges to $\hat{\gamma}^i$ when $k \in K$ goes to infinity (recall that $\|\nabla^c f(\hat{\gamma}^i)\| \neq 0$). Let N be an integer such that for any $k \geq N$ in K

$$\max_{B\in\mathcal{B}} \min_{b\in B} \nabla^c f(y^i_{k,j(k)})^t b \;<\; \frac{-2\eta}{3} \qquad \text{and} \qquad \max_{B\in\mathcal{B}} \min_{b\in B} \nabla^c f(z^i_k)^t b \;\geq\; \frac{-\eta}{3}.$$

Equation (11) and Theorem 4.6 which states that $\nabla^c f(\hat{z}^i) = 0$ guarantee the existence of N.

For $k \geq N$ in K, define the index $\ell(k) = \min\left\{\ell \geq j(k) : \min_{b\in B_k(y^i_{k,\ell})} \nabla^c f(y^i_{\ell,j})^t b \geq \frac{-\eta}{3}\right\}$. Therefore, Proposition 4.8 guarantees that $f(y^i_{k,j}) - f(y^i_{k,j+1}) \geq \sigma\|y^i_{k,j} - y^i_{k,j+1}\|$ when $j < \ell_{(k)}$ for $\sigma = \frac{\eta}{4\max\{\|b\| : b\in B\in\mathcal{B}\}}$. Writing out the telescopic sum leads to

$$\begin{aligned} f(y^i_{k,j(k)}) - f(y^i_{k,\ell(k)}) \;&=\; \sum_{j=j(k)}^{\ell(k)-1} \left(f(y^i_{k,j}) - f(y^i_{k,j+1})\right) \\ &\geq\; \sigma \sum_{j=j(k)}^{\ell(k)-1} \|y^i_{k,j} - y^i_{k,j+1}\| \;\geq\; \sigma\|y^i_{k,j(k)} - y^i_{k,\ell(k)}\|. \end{aligned}$$

Let $k \geq N$ in K be such that $f(y^i_{k,j(k)}) - f(y^i_{k,\ell(k)}) \leq \sigma\epsilon$ (by Proposition 4.7). Therefore, $\|y^i_{k,j(k)} - y^i_{k,\ell(k)}\| \leq \epsilon$. Equation (12) holds for $u = y^i_{k,j(k)}$, $v = y^i_{k,\ell(k)}$ and $B = B_k(y^i_{k,\ell(k)})$. It follows that

$$\begin{aligned}
\frac{-2\eta}{3} &> \min_{b\in B_k(y^i_{k,\ell(k)})} \nabla^c f(y^i_{k,j(k)})^t b \\
&= \min_{b\in B_k(y^i_{k,\ell(k)})} \nabla^c f(y^i_{k,\ell(k)})^t b + \left(\min_{b\in B_k(y^i_{k,\ell(k)})} \nabla^c f(y^i_{k,j(k)})^t b - \min_{b\in B_k(y^i_{k,\ell(k)})} \nabla^c f(y^i_{k,\ell(k)})^t b \right) \\
&> \frac{-\eta}{3} + \frac{-\eta}{3} \;=\; \frac{-2\eta}{3},
\end{aligned}$$

which is a contradiction. ■

In the next subsection, we illustrate the importance of the Strong Extended Extended Poll step on this last result through a small example. Without it, the algorithm produces a single accumulation point $\hat{x}$ for which there is a $\hat{y}$ in $\mathcal{N}(\hat{x})$ that satisfies $\nabla^c f(\hat{y}) \neq 0$.

4.4 Example

Consider the following example in which there are two continuous variable and a single binary one. In order to ease the notation, the continuous variables x^c are written $x^c = (u, v)$. The objective function is

$$f(x) = f(u, v, x^d) = g(u, v)(1 - x^d) + h(u, v)x^d.$$

where

$$g(u,v) = u^2 + v^2 \quad \text{and} \quad h(u,v) = \begin{cases} u^2 v + u(1-v) & \text{if } (u+v)^2 \leq 2 \\ u^2 v + u(1-v) + (u^2+v^2-2)^2 & \text{otherwise.} \end{cases}$$

The function h is partitioned as above to ensure that the level sets of f are bounded. It can be easily checked that both functions g and h are continuously differentiable.

The pattern search algorithm does not have a Search step. There is only a poll and an extended poll step. The objective function value parameter ξ that triggers the extended poll step is fixed at 1. The current mesh neighborhood at iteration k is defined to be

$$\mathcal{N}^c(x) = \{x + \Delta_k(0,1,0),\; x + \Delta_k(0,-1,0),\; x + \Delta_k(5,0,0),\; x + \Delta_k(-7,0,0)\}$$

for any $x = (u, v, x^d)$ except for $x = (2\Delta_k, 1 - \Delta_k, 1)$ in which case it is

$$\mathcal{N}^c(x) = \{x + \Delta_k(0,-1,0),\; x + \Delta_k(5,1,0),\; x + \Delta_k(-7,1,0)\}.$$

The set of neighbors of $x = (u, v, x^d)$ is $\mathcal{N}(x) = \{(u, v, 1 - x^d), (u, v, x^d)\}$. An iteration is declared successful and stops as soon as the incumbent is improved. When an iteration is unsuccessful, the mesh size parameter is divided by 2 otherwise it stays the same.

The algorithm is initiated at $x_0 = (1,0,0)$ with $\Delta_0 = \frac{1}{4}$ and with incumbent value $f(x_0) = 1$. The poll step evaluates the function at the points of $\mathcal{N}^c(x_0) : f(1,\frac{1}{4},0) = \frac{17}{16}, f(1,\frac{-1}{4},0) = \frac{17}{16}, f(\frac{9}{4},0,0) = \frac{81}{16}, f(\frac{-3}{4},0,0) = \frac{9}{16}$. This iteration is successful.

Iteration 1 is initiated at $x_1 = (\frac{-3}{4},0,0)$ with $\Delta_0 = \frac{1}{4}$ and $f(x_1) = \frac{9}{16}$. The poll step evaluates the function at three of the points of $\mathcal{N}^c(x_1) : f(\frac{-3}{4},\frac{1}{4},0) = \frac{10}{16}, f(\frac{-3}{4},\frac{-1}{4},0) = \frac{10}{16}, f(\frac{1}{2},0,0) = \frac{1}{4}$. This iteration is successful.

Iteration 2 is initiated at $x_2 = (\frac{1}{2},0,0)$ with $\Delta_0 = \frac{1}{4}$ and $f(x_2) = \frac{1}{4}$. The poll step evaluates the function at the points of $\mathcal{N}^c(x_2) : f(\frac{1}{2},\frac{1}{4},0) = \frac{5}{16}, f(\frac{1}{2},\frac{-1}{4},0) = \frac{5}{16}, f(\frac{7}{4},0,0) = \frac{49}{16}, f(\frac{-5}{4},0,0) = \frac{25}{16}$. Before declaring this iteration unsuccessful, polling must be conducted on the set of neighbors $\mathcal{N}(x_2) : f(\frac{1}{2},0,1) = \frac{1}{2}$. This value is within ξ of $f(x_2)$ and so extended polling must be conducted around this last point $y_{2,0}$ (the superscript i is dropped to ease the notation) The extended poll step finds $y_{2,1} = (\frac{1}{2},\frac{1}{4},1)$ in $\mathcal{N}^c(y_{2,0})$ with $f(y_{2,1}) = \frac{7}{16}$, then $y_{2,2} = (\frac{1}{2},\frac{1}{2},1)$ in $\mathcal{N}^c(y_{2,1})$ with $f(y_{2,1}) = \frac{3}{8}$, and $y_{2,3} = (\frac{1}{2},\frac{3}{4},1)$ in $\mathcal{N}^c(y_{2,2})$ with $f(y_{2,1}) = \frac{5}{16}$. It does not succeed in improving this last value in $\mathcal{N}^c(y_{2,3}) : f(\frac{1}{2},\frac{1}{2},1) = \frac{15}{32}, f(\frac{7}{4},1,1) = \frac{49}{16}, f(\frac{-5}{4},1,1) = \frac{25}{16}$. Therefore, iteration 2 is unsuccessful and iteration 3 starts at the same point $x_2 = (\frac{1}{2},0,0)$ with $\Delta_0 = \frac{1}{8}$ and $f(x_3) = \frac{1}{4}$.

Table 1 show that the algorithm generates cycles composed of two successful iterations, followed by an unsuccessful one. The three iterations detailed above, i.e., the first cycle, appear in the table by letting $a = \frac{1}{4}$. Iteration 3 initiates a new cycle with $a = \frac{1}{8}$.

Δ_k	x_k	$x_k + \Delta_k(0,1,0)$	$x_k + \Delta_k(0,-1,0)$	$x_k + \Delta_k(5,0,0)$	$x_k + \Delta_k(-7,0,0)$
a	$(4a,0,0)$ $16a^2$	$(4a,a,0)$ $17a^2$	$(4a,-a,0)$ $17a^2$	$(9a,0,0)$ $81a^2$	$(-3a,0,0)$ $9a^2$
a	$(-3a,0,0)$ $9a^2$	$(-3a,a,0)$ $10a^2$	$(-3a,-a,0)$ $10a^2$	$(-2a,0,0)$ $4a^2$	$(-10a,0,0)$ $100a^2$
a	$(2a,0,0)$ $4a^2$	$(2a,a,0)$ $5a^2$	$(2a,-a,0)$ $5a^2$	$(7a,0,0)$ $49a^2$	$(-5a,0,0)$ $25a^2$
	extended poll:	$y_k = y_{k,0}$	$y_{k,1}$	$y_{k,2}$ $\dots$	$z_k = y_{k,j_k^i}$
		$(2a,0,1)$ $2a$	$(2a,a,1)$ $2a(1-a(1-2a))$	$(2a,2a,1)$ $2a(1-2a(1-2a))$	$(2a,1-a,1)$ $2a^2(3-2a))$
	$\mathcal{N}^c(z_k)$:	$z_k + \Delta_k(0,-1,0)$	$z_k + \Delta_k(-7,1,0)$	$z_k + \Delta_k(5,1,0)$	
		$(2a,1-2a,1)$ $2a^2(4-4a^2)$	$(7a,1,1)$ $49a^2$	$(-5a,1,1)$ $25a^2$	

Table 1: In three iterations, the algorithm goes from $x_k = 4a, \Delta_k = a$ to $x_{k+3} = 2a, \Delta_{k+3} = \frac{a}{2}$

Figure 3 displays the iterates of the extended poll step from y_k to z_k. The circles represent the points $y_{k,j}$ for $j = 0,1,\dots j_k^i$. All these points are on the same line as the function decreases linearly when the variable u is fixed to $2a$. At the last point z_k, the current mesh

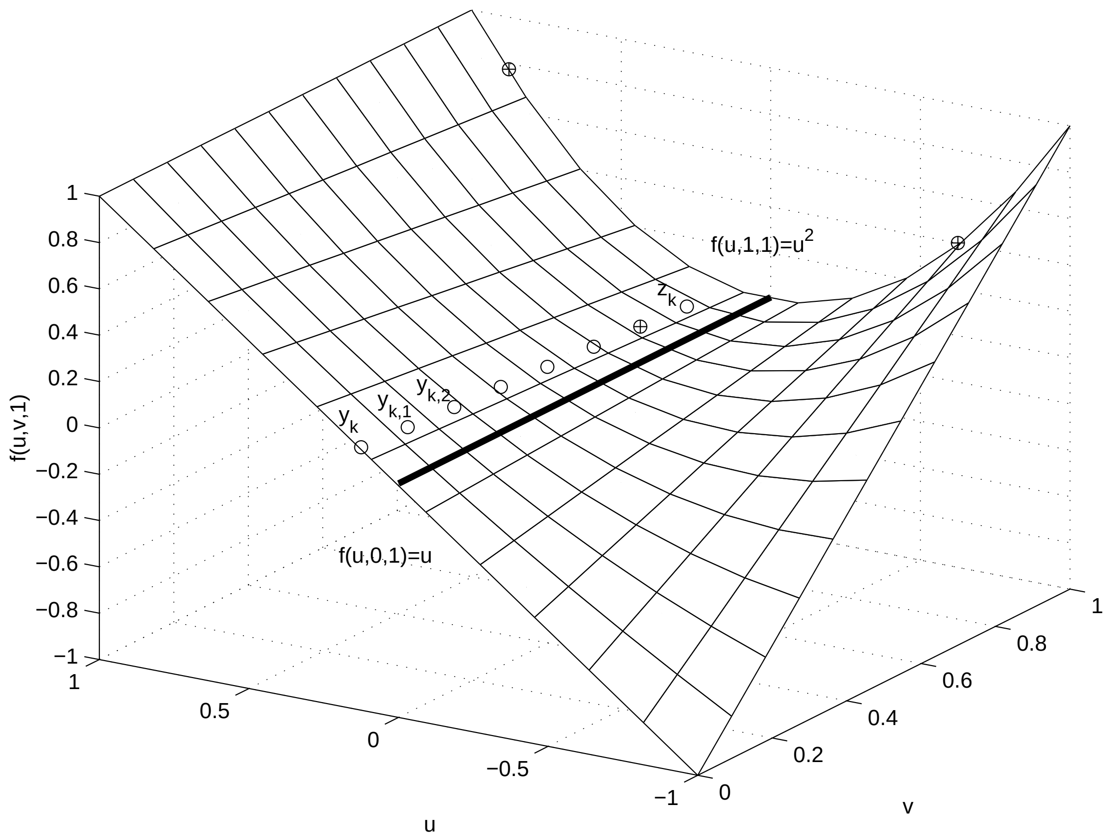


Figure 3: Extended poll from $y_k = (2a, 0, 1)$ to $z_k = (2a, 1 - a, 1)$

neighborhood is evaluated using a different positive basis. The set $\mathcal{N}^c(z_k)$ is represented by the three circled crosses.

As k goes to infinity, the set of points points $\{y_{k,j} : j = 0, 1, \ldots, j_k\}$ converges to the line segment from $\hat{y} = (0, 0, 1)$ to $\hat{z} = (0, 1, 1)$ which is represented by the thick line on Figure 3. The objective function value is equal to 0 there. The gradient norm is non-zero at $\hat{y}$ but decreases to zero at $\hat{z}$.

In order to ensure that the gradient norm is zero at all points of $\mathcal{N}(\hat{x})$, the stronger algorithm must be used. By doing this, the extended poll step at iteration 2 would discover the point $y_{2,1} = (\frac{-5}{4}, 0, 1)$ of $\mathcal{N}^c(y_{2,0})$ whose function value is $\frac{-5}{4}$. This iteration would be successful, and the iterates would eventually converge to the global minimizer of f.

Acknowledgments. The authors would like to thank David Applegate and Yin Zhang for discussions which helped improve the quality of this paper.

References

[1] Audet C. (1998), "Convergence Results for Pattern Search Algorithms are Tight," *TR98-24* Department of Computational & Applied Mathematics, Rice University, Houston TX.

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

[3] Booker A.J., Dennis J.E.Jr, Frank P.D., Serafini D.B., Torczon V. and Trosset M.W.(1998), "A Rigorous Framework for Optimization of Expensive Functions by Surrogates," *CRPC TR98739-s* Rice University, Houston TX. To appear in *Structural Optimization.*

[4] Conn A.R., Scheinberg K. and Toint Ph.L.(1997) "Recent progress in unconstrained nonlinear optimization without derivatives," *Mathematical Programming* 79, 397–414.

[5] Davis C.(1954), "Theory of positive linear dependence," *American Journal of Mathematics* 448–474.

[6] Dennis J.E.Jr and Torczon V.(1991), "Direct search methods on parallel machines," *SIAM Journal on Optimization* 1, 448–474.

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

[8] Lewis R.M. and Torczon V.(1996), "Pattern search algorithms for bound constrained minimization," *ICASE NASA Langley Research Center* TR 96-20. To appear in *SIAM Journal on Optimization.*

[9] Lewis R.M. and Torczon V.(1996), "Rank ordering and positive basis in pattern search algorithms," *ICASE NASA Langley Research Center* TR 96-71.

[10] Lewis R.M. and Torczon V.(1998), "Pattern search methods for linearly constrained minimization," *ICASE NASA Langley Research Center* TR 98-3.

[11] Torczon V.(1997), "On the Convergence of Pattern Search Algorithms," *SIAM Journal on Optimization* Vol.7 No.1, 1–25.