



## FILTER PATTERN SEARCH ALGORITHMS FOR MIXED VARIABLE CONSTRAINED OPTIMIZATION PROBLEMS

MARK A. ABRAMSON\*, CHARLES AUDET\*<sup>†</sup> AND J.E. DENNIS JR.\*

**Abstract:** A new class of algorithms for solving nonlinearly constrained mixed variable optimization problems is presented. This class combines and extends the Audet-Dennis generalized pattern search (GPS) algorithms for bound constrained mixed variable optimization, and their GPS-filter algorithms for general nonlinear constraints. In generalizing existing algorithms, new theoretical convergence results are presented that reduce seamlessly to existing results for more specific classes of problems. While no local continuity or smoothness assumptions are required to apply the algorithm, a hierarchy of theoretical convergence results that makes use of the Clarke calculus is given, in which local smoothness dictates what can be proved about certain limit points generated by the algorithm. We believe this is the first algorithm with provable convergence results to directly target this class of problems.

**Key words:** *optimization, pattern search algorithm, filter methods, mixed variable programming, categorical variables, nonsmooth optimization*

**Mathematics Subject Classification:** *65K05, 49M30, 90C30, 90C56*

### 1 Introduction

We analyze a new class of derivative-free filter pattern search algorithms for mixed variable optimization problems with general nonlinear constraints. This class of algorithms has already been shown in [2] to be effective on a problem from the engineering literature. That reference also shows the value of the mixed variable formulation over a basic parameter study.

Mixed variable optimization problems [6] are characterized by a mixture of continuous and categorical variables, the latter being discrete variables that must take on values from a predefined list or set of categories, or else the problem functions cannot be evaluated. Thus, continuous relaxations are not possible. These variables may be assigned numerical values for programming convenience, but these values are typically meaningless, and in fact, it will be generally misleading to think of defining a metric based on them. Type of material, color, and shape are common examples. Without loss of generality, we may represent categorical variables by their corresponding indices in the set that contains them (e.g., 1 = steel, 2 = aluminum, etc.).

\*The authors were supported by AFOSR F49620-01-1-0013, The Boeing Company, ExxonMobil Upstream Research Co. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, United States Government, or research sponsors.

<sup>†</sup>The second author was also supported by FCAR grant NC72792 and NSERC grant 239436-01.

In formulating the mixed variable programming (MVP) problem, we note that changes in the discrete variables can mean a change in the constraints. Thus, we denote  $n^c$  and  $n^d$  as the *maximum* dimensions of the continuous and discrete variables, respectively, and we partition each point  $x = (x^c, x^d)$  into continuous variables  $x^c \in X^c \subseteq \mathbb{R}^{n^c}$  and discrete variables  $x^d \in X^d \subseteq \mathbb{Z}^{n^d}$ . We adopt the convention of ignoring unused variables.

The problem under consideration can be expressed as follows:

$$\begin{aligned} \min_{x \in X} \quad & f(x) \\ \text{s.t.} \quad & C(x) \leq 0, \end{aligned} \tag{1.1}$$

where  $f : X \rightarrow \mathbb{R} \cup \{\infty\}$ , and  $C : X \rightarrow (\mathbb{R} \cup \{\infty\})^p$  with  $C = (C_1, C_2, \dots, C_p)^T$ . The domain  $X$  is the union of continuous domains across possible discrete variable values; i.e.,

$$X = \bigcup_{x^d \in X^d} (X^c(x^d) \times \{x^d\}),$$

with the convention that  $X = X^c$  if  $n^d = 0$ . Furthermore,  $X^c$  is defined by a finite set of bound and linear constraints, dependent on the values of  $x^d$ . That is,

$$X^c(x^d) = \{x^c \in \mathbb{R}^{n^c} : \ell(x^d) \leq A(x^d)x^c \leq u(x^d)\},$$

where  $A(x^d) \in \mathbb{R}^{m \times n^c}$  is a real matrix,  $\ell(x^d), u(x^d) \in (\mathbb{R} \cup \{\pm\infty\})^m$ , and  $\ell(x^d) \leq u(x^d)$  for all values of  $x^d$ . Note that this formulation is indeed a generalization of the standard NLP problem, in that, if  $n^d = 0$ , then the problem reduces to a standard NLP problem, in which  $\ell$ ,  $A$ , and  $u$  (and hence,  $X = X^c$ ) do not change. An important difference is that MVP problems will typically involve optimization on a nonmetric space.

Convergence in a mixed variable domain is assumed to be as one would expect: a sequence  $\{x_i\} = \{(x_i^c, x_i^d)\} \subset X$  converges to  $x = (x^c, x^d) \in X$  if  $x_i^c$  converges to  $x^c$  (under the standard definition) and  $x_i^d = x^d$  for all sufficiently large  $i$ .

Torczon [31] introduced the class of generalized pattern search (GPS) methods for solving unconstrained NLP problems, unifying a wide variety of existing derivative-free methods, and proving convergence of a subsequence of iterates to a stationary point, under the assumptions that all iterates lie in a compact set and that the objective function  $f$  is continuously differentiable in a neighborhood of the level set  $\mathcal{L}(x_0) = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$  defined by the initial point  $x_0 \in \mathbb{R}^n$ . Under similar assumptions, Lewis and Torczon extended pattern search to bound [23] and linearly constrained problems [24] by ensuring that directions used in the algorithm include tangent cone generators of all nearby constraints, thereby ensuring convergence of a subsequence of iterates to a Karush-Kuhn-Tucker (KKT) point. Lewis and Torczon [22] also establish the connection between pattern search and the positive basis theory of Davis [15], in which they generalize [31] to allow the use of any set of directions that positively span  $\mathbb{R}^n$  (i.e., directions such that any vector in  $\mathbb{R}^n$  can be expressed as a non-negative linear combination of these directions). This can significantly reduce the number of function evaluations.

Audet and Dennis [6] extended pattern search to bound constrained MVP problems under the assumption of continuous differentiability of the objective function on the neighborhood of a level set in which all iterates lie. The success of the method is demonstrated in [21] on a problem in the design of thermal insulation systems, an expanded version of which is discussed and numerically solved in [2] using the algorithm class given here. These algorithms were also successfully used in the quantitative reconstruction of objects from x-ray radiograph data [29]. A further extension to linearly constrained MVP problems with a stochastic objective function is given in [30].

A more general derivative-free framework for solving linearly constrained mixed variable problems is introduced in [27]. Instead of applying pattern search to the continuous variables, mathematical conditions are established, by which a suitably chosen derivative-free method could be used as a local continuous search and ensure convergence to a first-order stationary point. A general derivative-based approach for large-scale unconstrained MVP problems that exploits these conditions is given in [26].

An equivalent formulation of GPS for linearly constrained NLP problems was introduced and analyzed by Audet and Dennis [7] for functions that are less well-behaved. They apply the nonsmooth calculus of Clarke [13] to establish convergence properties for functions lacking the smoothness properties of those studied in previous work. In doing so, they present a hierarchy of convergence results for bound and linearly constrained problems, in which the strength of the results depends on local continuity and smoothness conditions of the objective function. As a consequence, they establish some of the earlier results of [31], [23], and [24] as corollary to theirs with much shorter and simpler proofs.

For NLP problems with general nonlinear constraints, Lewis and Torczon [25] apply bound constrained pattern search to an augmented Lagrangian function [14] and show that, under the same assumptions as in [14], plus a mild restriction on search directions, the algorithm converges to a KKT first-order stationary point. It is unclear to us how to choose the parameters needed to implement their algorithm. The algorithm given here has no such difficulty.

Audet and Dennis [8] adapt a filter method within the GPS framework to handle general nonlinear constraints. Originally introduced by Fletcher and Leyffer [17] to conveniently globalize sequential quadratic programming (SQP) and sequential linear programming (SLP), filter methods accept steps if either the objective function or an aggregate constraint violation function is reduced. Fletcher, Leyffer, and Toint [18] show convergence of the SLP-based approach to a limit point satisfying Fritz John [20] optimality conditions; they show convergence of the SQP approach to a KKT point [19], provided a constraint qualification is satisfied. However, in both cases, more than a simple decrease in the function values is required for convergence with these properties.

Audet and Dennis show convergence to limit points  $\hat{x}$  having almost the same characterization as in [7], but with only a simple decrease in the objective or constraint violation function required. While they are unable to show convergence to a point satisfying KKT optimality conditions (and, in fact, have counterexamples [8]), in that  $-\nabla f(\hat{x})$  does not necessarily belong to the normal cone at  $\hat{x}$ , they are able to show that  $-\nabla f(\hat{x})$  belongs to the polar of a cone defined by directions that are used infinitely often. Thus, a richer set of directions, although more costly, will increase the likelihood of achieving convergence to a KKT point.

The present paper introduces a filter GPS algorithm for MVP problems with general nonlinear constraints. In doing so, we make use of the nonsmooth Clarke [13] calculus as in [7] and [8] to establish a unifying hierarchy of results for all the pattern search methods to date.

The paper is outlined as follows. After presenting some basic ideas on mixed variables in Section 2, we construct the mixed variable GPS (MVPS) method of Audet and Dennis [6] in Section 3, retailored for linearly constrained MVP problems. In Section 4, we extend this development to general constraints by incorporating a filter and we introduce the Filter-MVPS algorithm. We establish theoretical convergence properties for the new algorithm in Section 5.

**Notation.**  $\mathbb{R}$ ,  $\mathbb{Z}$ , and  $\mathbb{Z}_+$  denote the set of real numbers, integers, and nonnegative integers, respectively. For any finite set  $S \subset \mathbb{R}^n$ , we may refer to the matrix  $S$  as the one

whose columns are the elements of  $S$ . Similarly, for any matrix  $A$ , the notation  $a \in A$  means that  $a$  is a column of  $A$ . When  $x$  has only continuous variables, we denote by  $B(x, \varepsilon)$  the ball of radius  $\varepsilon > 0$  centered at  $x$ .

## 2 Local Optimality for Mixed Variables

In order to solve problems with categorical variables, a notion of local optimality is needed, since we are not dealing with a metric space. For continuous variables, this is well-defined in terms of local neighborhoods. However, for categorical variables, a local neighborhood must be defined by the user, and there may be no obvious choice for doing so; special knowledge of the underlying engineering process or physical problem may be the only guide.

To keep the definition as general as possible, we define local neighborhoods in terms of a set-valued function  $\mathcal{N} : X \rightarrow 2^X$ , where  $2^X$  denotes the power set (or set of all possible subsets of  $X$ ). By convention, we assume that for all  $x \in X$ , the set  $\mathcal{N}(x)$  is finite, and  $x \in \mathcal{N}(x)$ . As an example, one common choice of neighborhood function for integer variables is the one defined by  $\mathcal{N}(x) = \{y \in X : y^c = x^c, \|y^d - x^d\|_1 \leq 1\}$ . However, categorical variables may have no inherent metric, which would make this particular choice inapplicable.

With this construction, the classical definition of local optimality is extended to mixed variable domains by the following definition, which is similar to one found in [6].

**Definition 2.1.** Let  $\Omega = \{x \in X : C(x) \leq 0\}$ . A point  $x = (x^c, x^d) \in X$  is said to be a *local minimizer of  $f$  on  $\Omega$  with respect to the set of neighbors  $\mathcal{N}(x) \subset X$*  if there exists an  $\varepsilon > 0$  such that  $f(x) \leq f(v)$  for all  $v$  in the set

$$\Omega \cap \bigcup_{y \in \mathcal{N}(x)} (B(y^c, \varepsilon) \times \{y^d\}). \quad (2.1)$$

## 3 Pattern Search for Linearly Constrained MVPs

In order to introduce the Filter-MVPS algorithm, it is helpful to first build up the structure by describing the GPS algorithm for linearly constrained MVP problems. Most of the discussion in this section comes from [6], but some improvements are added here, including a slightly more general mesh construction and the treatment of linear constraints and functions that are not necessarily continuously differentiable.

A pattern search algorithm is characterized by a sequence of feasible iterates  $\{x_k\} \subset X$  with nonincreasing objective function values. Each iteration consists of two key steps – an optional SEARCH step and a local POLL step – in which the objective function is evaluated at a finite number of points (called trial points) lying on a carefully constructed mesh (to be formally defined for MVP problems later) in an attempt to find a new iterate with a lower objective function value than the current iterate (called the incumbent).

A key practical point in the Audet-Dennis version of the GPS algorithm is that they explicitly separate out the SEARCH step from the POLL step within the iteration. In the SEARCH step, any finitely terminating strategy may be used in selecting a finite number of trial points, as long as the points lie on the mesh. This flexibility lends itself quite easily to hybrid algorithms and enables the user to apply specialized knowledge of the problem. The user can apply a few steps of a favorite heuristic, such as random sampling, simulated annealing, a few generations of a genetic algorithm, etc., or perhaps use a few steps of an algorithm to optimize an inexpensive surrogate function on the mesh, as is common

in difficult engineering design problems with expensive function evaluations [5, 10, 11, 12]. While the SEARCH step contributes nothing to the convergence theory of GPS (and in fact, an unsuitable SEARCH may impede performance), the use of surrogates enables the user to potentially gain significant improvement early on in the iteration process at much lower cost. See [28] for evidence to support this assertion.

If the SEARCH step fails to find an *improved mesh point* (i.e., a point with lower objective function value), then the POLL step is invoked, in which the function is evaluated at a finite set of neighboring mesh points around the incumbent, called the *poll set*. The POLL step is more carefully structured, so as to help ensure the algorithm’s theoretical convergence properties. If either the SEARCH or POLL step finds an improved mesh point, then it becomes the incumbent, and the mesh is retained or coarsened. If no improved mesh point is found, then  $x_k$  is said to be a *mesh local optimizer*, and the current mesh is refined. The next iteration then begins on a finer mesh.

**3.1 Construction of the Mesh and Poll Set**

The following construction is slightly more general than in [6]. For each combination  $i = 1, 2, \dots, i_{\max}$ , of values that the discrete variables may possibly take, a set of positive spanning directions  $D^i$  in  $\mathbb{R}^{n^c}$  is constructed by forming the product

$$D^i = G_i Z_i, \tag{3.1}$$

where  $G_i \in \mathbb{R}^{n^c \times n^c}$  is a nonsingular generating matrix, and  $Z_i \in \mathbb{Z}^{n^c \times |D^i|}$ . We will sometimes use  $D(x)$  in place of  $D^i$  to indicate that the set of directions is associated with the discrete variable values of  $x \in X$ . The set  $D$  is then defined by  $D = \bigcup_{i=1}^{i_{\max}} D^i$ .

The mesh  $M_k$  at iteration  $k$  is formed as the direct product of  $X^d$  with the union of a finite number of lattices in  $X^c$ , each of which is the union of lattices centered at the continuous part of the variables at all previously evaluated trial points  $S_k$ , where  $S_0$  is the set of initial points. More precisely,

$$M_k = \bigcup_{i=1}^{i_{\max}} M_k^i \times X^d \tag{3.2}$$

with  $M_k^i = \bigcup_{x \in S_k} \{x^c + \Delta_k D^i z : z \in \mathbb{Z}_+^{|D^i|}\} \subset \mathbb{R}^{n^c}$ ,

where  $\Delta_k > 0$  is the mesh size parameter. The mesh is purely conceptual and is never explicitly created; instead, mesh points are generated as necessary in the algorithm.

Using this construction, we also require that the neighborhood function  $\mathcal{N}$  be constructed so that all discrete neighbors of the current iterate lie on the current mesh; i.e.,  $\mathcal{N}(x_k) \subseteq M_k$  for all  $k$ . Also observe that each lattice in (3.2) is expressed as a translation from  $x_k^c$ , as opposed to  $y_k^c$ , for some  $y_k \in \mathcal{N}(x_k)$ . This is necessary to ensure convergence of the algorithm, but it does not mean that a point and its discrete neighbors have the same continuous variable values. In fact, Kokkolaras *et al.* [21] construct their neighbor sets in a way that neighbors often do *not* have the same continuous variable values.

Polling in the MVPS algorithm is performed with respect to the continuous variables, the discrete neighbor points, and the set of points generated by an EXTENDED POLL step. At iteration  $k$ , let  $D_k(x) \subseteq D^{i_0} \subseteq D$  denote the positive spanning set of poll directions for some  $x \in S_k$  corresponding to the  $i_0$ -th set of discrete variable values. The *poll set* centered at  $x$  is defined as

$$P_k(x) = \{x\} \cup \{x + \Delta_k(d, 0) \in X : d \in D_k(x)\} \subset M_k. \tag{3.3}$$

The notation  $(d, 0)$  is consistent with the partitioning into continuous and discrete variables, respectively, where 0 means that discrete variables do not change value. Thus,  $x + \Delta_k(d, 0) = (x^c + \Delta_k d, x_k^d)$ .

Whenever poll set and set of discrete neighbors fail to produce a lower objective function value, MVPS performs an EXTENDED POLL step, in which additional polling is performed around any promising points in the set of discrete neighbors whose objective function value is sufficiently close to the incumbent value. That is, if  $y \in \mathcal{N}(x_k)$  satisfies  $f(x_k) \leq f(y) < f(x_k) + \xi_k$  for some user-specified tolerance value  $\xi_k \geq \xi$  (called the *extended poll trigger*), where  $\xi$  is a fixed positive scalar, then we begin a finite sequence of POLL steps about the points  $\{y_k^j\}_{j=1}^{J_k}$ , beginning with  $y_k^0 = y_k \in \mathcal{N}(x_k)$  and ending when either  $f(y_k^{J_k} + \Delta_k(d, 0)) < f(x_k)$  for some  $d \in D_k(y_k^{J_k})$ , or when  $f(x_k) \leq f(y_k^{J_k} + \Delta_k(d, 0))$  for all  $d \in D_k(y_k^{J_k})$ . For this discussion, we let  $z_k = y_k^{J_k}$ , the last iterate (or *endpoint*) of the EXTENDED POLL step. We will show in Section 5 that under reasonable assumptions  $J_k$  is always finite.

We should note that in practice, the parameter  $\xi_k$  is typically set as a percentage of the objective function value (but bounded away from zero), such as, say,  $\xi_k = \max\{\xi, 0.05|f(x_k)|\}$ . A relatively high choice of  $\xi_k$  will generate more EXTENDED POLL steps, which is likely to lead to a better local solution, but at a cost of more function evaluations. On the other hand, a lower value of  $\xi_k$  will require fewer function evaluations, but it will probably result in a poorer quality local solution.

The set of extended poll points for a discrete neighbor  $y \in \mathcal{N}(x_k)$ , denoted  $\mathcal{E}(y)$ , contains a subset of the points  $\{P_k(y_k^j)\}_{j=1}^{J_k}$ . At iteration  $k$ , the set of points evaluated in the EXTENDED POLL step (or *extended poll set*) is given by

$$\mathcal{X}_k(\xi_k) = \bigcup_{y \in \mathcal{N}_k^{\xi_k}} \mathcal{E}(y), \quad (3.4)$$

where  $\mathcal{N}_k^{\xi_k} = \{y \in \mathcal{N}(x_k) : f(x_k) \leq f(y) \leq f(x_k) + \xi_k\}$ .

### 3.2 Update Rules

If either the SEARCH, POLL, or EXTENDED POLL step is successful at finding an improved mesh point, then it becomes the new incumbent  $x_{k+1}$ , and the mesh may be retained or coarsened according to the rule,

$$\Delta_{k+1} = \tau^{m_k^+} \Delta_k, \quad (3.5)$$

where  $\tau > 1$  is rational and fixed over all iterations, and the integer  $m_k^+$  satisfies  $0 \leq m_k^+ \leq m_{\max}$  for some fixed integer  $m_{\max} \geq 0$ . Coarsening of the mesh does not prevent convergence of the algorithm, and may make it faster. Note that only a simple decrease in the objective function value is required.

If the SEARCH, POLL, and EXTENDED POLL steps all fail to find an improved mesh point, then the incumbent is a mesh local optimizer and remains unchanged (or, alternatively, can be chosen as a point having the same function value as the incumbent, if one exists), while the mesh is refined according to the rule,

$$\Delta_{k+1} = \tau^{m_k^-} \Delta_k, \quad (3.6)$$

where  $\tau > 1$  is defined above,  $\tau^{m_k^-} \in (0, 1)$ , and the integer  $m_k^-$  satisfies  $m_{\min} \leq m_k^- \leq -1$  for some fixed integer  $m_{\min}$ . It follows that, for any integer  $k \geq 0$ , there exists an integer  $r_k$  such that

$$\Delta_k = \tau^{r_k} \Delta_0. \quad (3.7)$$

**3.3 Linear Constraints**

For linear constraints, infeasible points are simply discarded without being evaluated. However, to ensure that appropriate convergence results are retained, the directions that define the mesh be sufficiently rich to ensure that polling directions can be chosen that conform to the geometry of the constraint boundaries, and these directions must be used in infinitely many iterations. For our analysis, we need the following definition (from [7]), which abstracts this notion of conformity.

**Definition 3.1.** A rule for selecting the positive spanning sets  $D_k(x) \subseteq D$  conforms to  $X \subset \mathbb{R}^n$  at  $x \in \mathbb{R}^n$  for some specified  $\epsilon > 0$ , if at each iteration  $k$  and for each  $y$  in the boundary of  $X$  for which  $\|y - x\| < \epsilon$ , the tangent cone  $T_X(y)$  is generated by nonnegative linear combinations of a subset of the columns of  $D_k(x)$ .

Thus, just as in [7] and [24], in order to ensure convergence, we require that the rule for selecting directions always conforms to  $X^\epsilon$  for some  $\epsilon > 0$ . To enforce this condition, we appeal to the construction of Lewis and Torczon [24], who provide an algorithm for choosing conforming directions using standard linear algebra tools.

Nonlinear constraints pose a problem for GPS algorithms in that choosing enough directions to conform to the geometry of the constraints (to guarantee convergence to a KKT point) would require an infinite number of directions in  $D$ , which the convergence theory does not support. Thus, a different strategy must be employed to handle nonlinear constraints. In the next section, we add a filter to do this.

**4 The Filter-MVPS Algorithm**

In filter algorithms, the goal is to minimize two functions, the objective  $f$  and a continuous aggregate constraint violation function  $h$  that satisfies  $h(x) \geq 0$  with  $h(x) = 0$  if and only if  $x$  is feasible. Specifically, we will set

$$h(x) = \|C(x)_+\|_2^2, \tag{4.1}$$

where  $C(x)_+$  is the vector of constraint violations at  $x$ ; i.e., for  $i = 1, 2, \dots, m$ ,  $C_i(x)_+ = C_i(x)$  if  $C_i(x) > 0$ ; otherwise,  $C_i(x)_+ = 0$ . Our reason for making this particular choice is that  $h$  inherits whatever smoothness properties  $C$  possesses [8]. Other choices for  $h$  may be used, but convergence properties may be weakened.

In our case, and consistent with [8], we define a second constraint violation function  $h_X = h + \psi_X$ , where  $\psi_X$  is the indicator function for  $X$ . It is 0 on  $X$  and  $+\infty$  elsewhere. This construct is convenient. Again we point out that there is no difficulty in using this convenient construct since we simply discard any  $x \notin X$  without evaluating its constraint violation. We will see in Section 5 that convergence results will depend on the smoothness of  $h$  and not  $h_X$ .

The Filter-MVPS algorithm can be viewed as either an extension of the Filter-GPS algorithm [8] for mixed variables, or as an extension of the mixed variable GPS algorithm of Audet and Dennis [6] for general nonlinear constraints. We present it here as the latter, and appeal to [8] for the construction of the filter.

**4.1 Filters**

The definition of dominance provided below, which comes from the multi-criteria optimization literature, is adapted from a similar term in [17], so that it is defined with respect to

the objective function  $f$  and constraint violation function  $h$ . This adaptation is consistent with [8]. A formal definition of a filter follows immediately thereafter.

**Definition 4.1.** A point  $x \in \mathbb{R}^n$  is said to *dominate*  $y \in \mathbb{R}^n$ , written  $x \prec y$ , if  $f(x) \leq f(y)$  and  $h_X(x) \leq h_X(y)$  with either  $f(x) < f(y)$  or  $h_X(x) < h_X(y)$ .

**Definition 4.2.** A *filter*, denoted  $\mathcal{F}$ , is a finite set of points in the domain of  $f$  and  $h$  such that no pair of points  $x$  and  $y$  in the set have the relation  $x \prec y$ .

In constructing a filter for GPS algorithms, we put two additional restrictions on  $\mathcal{F}$ . First, we set a bound  $h_{\max}$  on aggregate constraint violation, so that each point  $y \in \mathcal{F}$  satisfies  $h_X(y) < h_{\max}$ . Second, we include only infeasible points in the filter and track feasible points separately, consistent with [17, 16]. With these two modifications, the following terminology is now provided.

**Definition 4.3.** A point  $x$  is said to be *filtered* by a filter  $\mathcal{F}$  if any of the following properties hold:

1. There exists a point  $y \in \mathcal{F}$  such that  $y \preceq x$ ,
2.  $h_X(x) \geq h_{\max}$ ,
3.  $h_X(x) = 0$  and  $f(x) \geq f^F$ , where  $f^F$  is the objective function value of the best feasible point found thus far.

The point  $x$  is said to be *unfiltered* by  $\mathcal{F}$  if it is not filtered by  $\mathcal{F}$ .

Thus, the set of filtered points, denoted by  $\overline{\mathcal{F}}$ , is given by

$$\overline{\mathcal{F}} = \bigcup_{x \in \mathcal{F}} \{y : y \succeq x\} \cup \{y : h_X(y) \geq h_{\max}\} \cup \{y : h_X(y) = 0, f(y) \geq f^F\}. \quad (4.2)$$

Observe that, with this notation, if a new trial point has the same function values as those of any point in the filter, then the trial point is filtered. Thus, only the first point with such values is accepted into the filter.

#### 4.2 Description of the Algorithm

For the new class of algorithms, we change the notation slightly. At each iteration  $k$ , an iterate  $x_k$  is not necessarily the point around which the POLL step will be performed. Instead, the filter  $\mathcal{F}_k$  is updated at each iteration  $k$  (this requires an initial population of the filter  $\mathcal{F}_0$ ), and the *poll center*  $p_k$  is chosen as either the incumbent best feasible point  $p_k^F$  or the incumbent least infeasible point  $p_k^I$ . Neither choice for  $p_k$  will affect the convergence theory, but we have had good numerical experience using both points during a run, and not using  $p_k^I$  is essentially the same as not using the filter. For a given poll center  $p_k$ , the poll set  $P_k(p_k)$  is defined in (3.3).

Because the filter seeks a better point with respect to either of the two functions (the objective function  $f$  and the constraint violation function  $h_X$ ), a change must be made to the rule for selecting discrete neighbors, about which to perform an EXTENDED POLL step. Recall that in the MVPS algorithm, extended polling is performed around any discrete neighbor whose objective function value is sufficiently close to that of the current iterate (i.e., “almost” an improved mesh point). With the addition of nonlinear constraints to the problem, we require a notion of a discrete neighbor “almost” generating a new incumbent best feasible point or least infeasible point.



While this issue has by no means a single workable approach, the implementation here has the desirable property of being a generalization of the MVPS algorithm. At iteration  $k$ , let  $f_k^F = f(p_k^F)$  denote the objective function value of the incumbent best feasible point. If no feasible point exists, we set  $f_k^F = \infty$ . Similarly, let  $h_k^I = h_X(p_k^I) > 0$  be the constraint violation function value of the incumbent least infeasible point. If no such point exists, we set  $h_k^I = h_{\max}$  and  $f_k^I = -\infty$ , where  $f_k^I = f(p_k^I)$  is the objective function value of the least infeasible point. Given current poll center  $p_k$  and user-specified extended poll triggers  $\xi_k^f \leq \xi > 0$  and  $\xi_k^h \leq \xi > 0$  for  $f$  and  $h$ , respectively (where  $\xi$  is a positive constant), we perform an EXTENDED POLL step around any discrete neighbor  $y_k \in \mathcal{N}(p_k)$  satisfying either  $0 < h_k^I < h_X(y_k) < \min(h_k^I + \xi_k^h, h_{\max})$ , or  $h_X(y_k) = 0$  with  $f_k^F < f(y_k) < f_k^F + \xi_k^f$ . The extended poll triggers  $\xi_k^f$  and  $\xi_k^h$  can also be set according to the categorical variable values associated with the current poll center, but this dependency is not included in the notation, so as not to obfuscate the ideas presented here.

Similar to the MVPS algorithm described in Section 3, the EXTENDED POLL STEP generates a sequence of EXTENDED POLL centers  $\{y_k^j\}_{j=0}^{J_k}$ , beginning with  $y_k^0 = y_k$  and ending with *extended poll endpoint*,  $y_k^{J_k} = z_k$ .

Thus, at iteration  $k$ , the set of all points evaluated in the EXTENDED POLL step, denoted  $\mathcal{X}_k(\xi_k^f, \xi_k^h)$ , is

$$\mathcal{X}_k(\xi_k^f, \xi_k^h) = \bigcup_{y \in \mathcal{N}_k^f \cup \mathcal{N}_k^h} \mathcal{E}(y) \tag{4.3}$$

where  $\mathcal{E}(y)$  denotes the set of extended poll points, and

$$\mathcal{N}_k^f = \{y \in \mathcal{N}(p_k) : h_X(y) = 0, f_k^F \leq f(y) \leq f_k^F + \xi_k^f\}, \tag{4.4}$$

$$\mathcal{N}_k^h = \{y \in \mathcal{N}(p_k) : 0 < h_k^I < h_X(y) < \min(h_k^I + \xi_k^h, h_{\max})\}. \tag{4.5}$$

The set of trial points is defined as  $T_k = S_k \cup P_k(p_k) \cup \mathcal{N}(p_k) \cup \mathcal{X}_k(\xi_k^f, \xi_k^h)$ , where  $S_k$  is the finite set of mesh points evaluated during the SEARCH step.

The addition of the filter complicates our notions of success or failure of the iteration in finding a desirable iterate. The following definitions now define the two outcomes of the SEARCH, POLL, and EXTENDED POLL steps.

**Definition 4.4.** Let  $T_k$  denote the set of trial points to be evaluated at iteration  $k$ , and let  $\overline{\mathcal{F}}_k$  denote the set of filtered points described by (4.2). A point  $y \in T_k$  is said to be an *unfiltered point* if  $y \notin \overline{\mathcal{F}}_k$ .

**Definition 4.5.** Let  $P_k(p_k)$  denote the poll set centered at the point  $p_k$ , and let  $\overline{\mathcal{F}}_k$  denote the set of filtered points described by (4.2). The point  $p_k$  is said to be a *mesh isolated filter point* if all mesh points evaluated during the POLL and EXTENDED POLL steps are filtered; i.e.,  $P_k(p_k) \cup \mathcal{N}(p_k) \cup \mathcal{X}(\xi_k^f, \xi_k^h) \subset \overline{\mathcal{F}}_k$ .

Figure 1 is a depiction of a filter on a bi-loss graph, in which the best feasible and least infeasible solutions are indicated, and the feasible solutions lie on the vertical axis (labeled  $f$ ). Dashed lines indicate the areas for which an EXTENDED POLL step is triggered. If a feasible discrete neighbor has an objective function value that lies on  $(f_k^F, f_k^F + \xi_k^f)$  (i.e., higher on the axis than the current feasible incumbent, but lower than the horizontal dashed line), an EXTENDED POLL step is performed around this discrete neighbor. Similarly, an EXTENDED POLL step is performed if an infeasible discrete neighbor has a constraint violation function value that lies on  $(h_k^I, h_k^I + \xi_k^h)$  (i.e., it lies to the right of the current least infeasible solution, but left of the vertical dashed line).

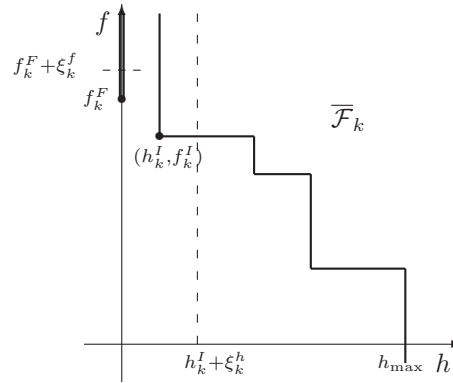


Figure 1: MVP Filter and Extended Poll Triggers.

The goal of each iteration is to find an unfiltered point, but the details of when to continue an EXTENDED POLL step must be generalized from the simple decrease condition in  $f$  under which the MVPS algorithm operates. More specifically, if the EXTENDED POLL step finds an unfiltered point, it is added to the filter, the best feasible and least infeasible points are updated (if appropriate), the new poll center is chosen, and the mesh is coarsened according to the rule in (3.5). If the EXTENDED POLL step fails to find a new point  $y$  satisfying  $y \in \mathcal{N}_k^f \cup \mathcal{N}_k^h$ , then the current incumbent poll center  $p_k$  is declared to be a mesh isolated filter point, the next poll center is chosen from among the current best feasible and least infeasible points (which remain unchanged), and the mesh is refined according to the rule in (3.6).

Finally, we treat the case in which extended poll points are filtered, yet still belong to  $\mathcal{N}_k^f$  or  $\mathcal{N}_k^h$ . To do so, we establish the notion of a temporary *local filter*. At iteration  $k$ , for each discrete neighbor  $y_k$ , a local filter  $\mathcal{F}_k^L(y_k)$  is constructed relative to the current EXTENDED POLL step and initialized only with the point  $y_k$  and  $h_{\max}^L = \min(h_k^I + \xi_k^h, h_{\max})$ . As with the MVPS algorithm, the extended poll sequence  $\{y_k^j\}_{j=1}^{J_k}$  begins with  $y_k^0 = y_k$  and ends with  $z_k = y_k^{J_k}$ , where each  $y_k^j$  is the poll center of the local filter – chosen either as the best feasible or least infeasible point, relative to the local filter. Extended polling with respect to  $y_k$  proceeds, with the local filter being updated as appropriate, until no more unfiltered mesh points can be found with respect to the new local filter, or until an unfiltered point is found with respect to the main filter. When either of these conditions is satisfied, the EXTENDED POLL step ends, and the main filter is appropriately updated with the points of the local filter, which is then discarded. The mesh size parameter  $\Delta_k$ , which is constant throughout the step, is then updated, depending on whether an unfiltered point (with respect to the main filter) has been found.

The EXTENDED POLL step and Filter-MVPS (FMVPS) Algorithm are summarized in Figures 2 and 3. In [2], the algorithm was applied to the design of a load-bearing thermal insulation system, in which case, the resulting design achieved a 50% improvement over previous designs.

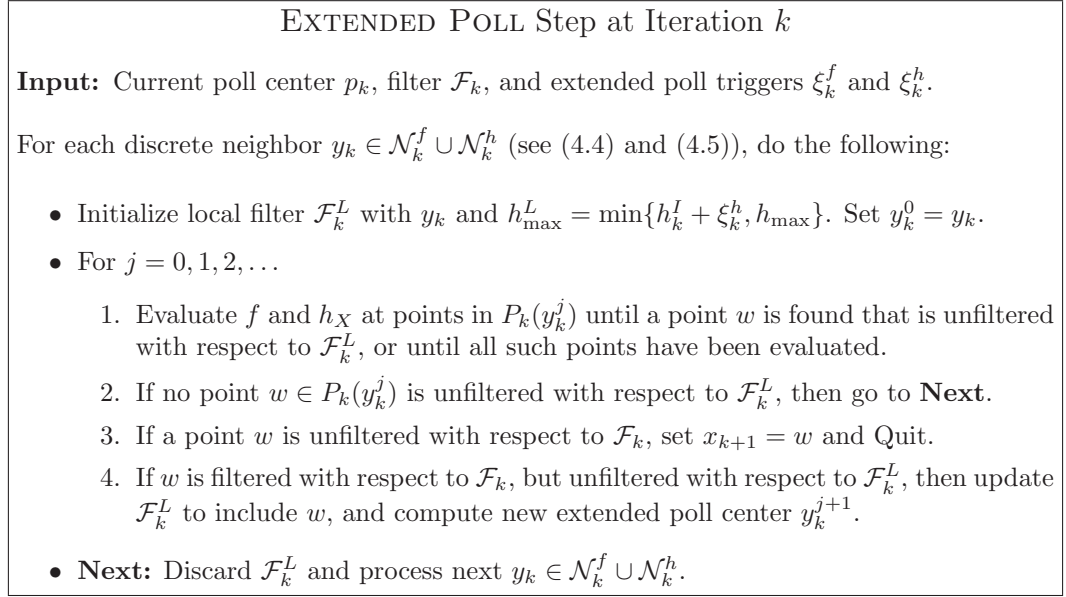


Figure 2: Extended Poll Step for the FMVPS Algorithm

## 5 Convergence Analysis

The convergence properties of the new algorithm are now presented. First, the behavior of the mesh size parameter  $\Delta_k$  will be shown to have the same behavior as in previous algorithms, and a general characterization of limit points of certain subsequences is given. Results for the constraint violation function and for the objective function follow, similar to those found in [8]. Finally, stronger results for a more specific implementation of the new algorithm are provided. These mimic those found in [6], but apply to the more general MVP problem with nonlinear constraints. We should note that many of the results presented here are significantly different than the original presentation in [1].

We make the following assumptions, consistent with those of previous GPS algorithms:

**A1:** All iterates  $\{x_k\}$  produced by the algorithm lie in a compact set.

**A2:** For each fixed  $x^d$ , the corresponding set of directions  $D^i = G_i Z_i$ , as defined in (3.1), includes tangent cone generators for every point in  $X^c(x^d)$ .

Assumption A1 is satisfied if the level sets of  $f$  are bounded. Since the EXTENDED POLL step is conducted for a fixed value of  $\Delta_k$ , Assumption A1 ensures that the index  $J_k$  of the EXTENDED POLL endpoint is finite; thus the algorithm is well-defined. Assumption A2 is actually a condition on the linear constraints. The structure of the coefficient matrix  $A$  determines whether or not it is possible to construct a set of directions that both satisfies (3.1) and includes all of the tangent cone generators. In [24], a more specific assumption is made, in which  $G_i$  would be the identity matrix, and  $A$  has only rational entries.

### 5.1 Mesh Size Behavior and Limit Points

The behavior of the mesh size was originally characterized for unconstrained problems by Torczon [31], independent of the smoothness of the objective function. It was extended to

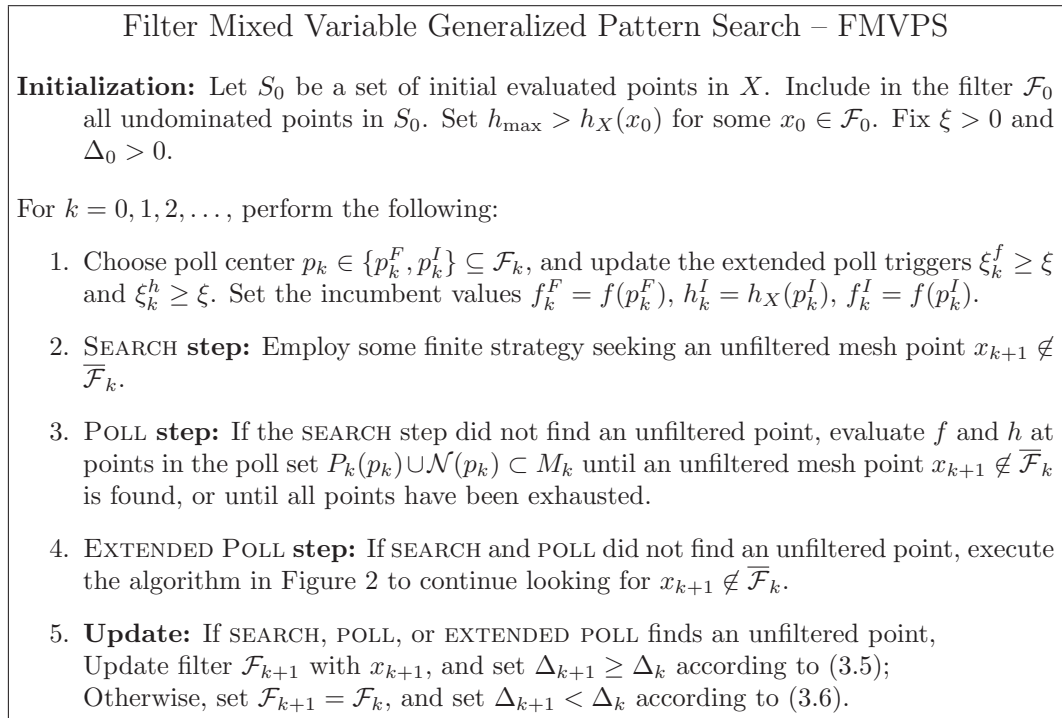


Figure 3: FMVPS Algorithm

MVP problems by Audet and Dennis [6], who later adapted the proof to provide a lower bound on the distance between mesh points at each iteration [7]. The proofs here are straightforward extensions of the latter work to MVP problems. The first lemma provides the lower bound on the distance between any two mesh points whose continuous variable values do not coincide, while the second lemma shows that the mesh size parameter is bounded above. The theorem that follows shows the key result that  $\liminf_{k \rightarrow +\infty} \Delta_k = 0$ .

**Lemma 5.1.** For any integer  $k \geq 0$ , let  $u$  and  $v$  be any distinct points in the mesh  $M_k$  such that  $u^d = v^d$ . Then for any norm for which all nonzero integer vectors have norm at least 1,

$$\|u^c - v^c\| \geq \frac{\Delta_k}{\|G_i^{-1}\|}.$$

where the index  $i$  corresponds to the discrete variable values of  $u$  and  $v$ .

*Proof.* Using (3.2), we let  $u^c = x_k^c + \Delta_k D^i z_u$  and  $v^c = x_k^c + \Delta_k D^i z_v$  define the continuous part of two distinct points on  $M_k$  with both  $z_u, z_v \in \mathbb{Z}_+^{|D^i|}$ . Furthermore, since we assume that  $u$  and  $v$  are distinct with  $u^d = v^d$ , we must have  $u^c \neq v^c$ , and thus  $z_u \neq z_v$ . Then

$$\|u^c - v^c\| = \Delta_k \|D^i(z_u - z_v)\| = \Delta_k \|G_i Z_i(z_u - z_v)\| \geq \Delta_k \frac{\|Z_i(z_u - z_v)\|}{\|G_i^{-1}\|} \geq \frac{\Delta_k}{\|G_i^{-1}\|},$$

the last inequality because  $Z_i(z_u - z_v)$  is a nonzero integer vector with norm greater than or equal to one.  $\square$

**Lemma 5.2.** There exists a positive integer  $r^u$  such that  $\Delta_k \leq \Delta_0 \tau^{r^u}$  for any integer  $k \geq 0$ .

*Proof.* Under Assumption A1, the discrete variables can only take on a finite number of values in  $L_X(x_0)$ . Let  $i_{\max}$  denote this number, and let  $I = \{1, 2, \dots, i_{\max}\}$ . Also under Assumption A1, for each  $i \in I$ , let  $Y_i$  be a compact set in  $\mathbb{R}^{n^c}$  containing all GPS iterates whose discrete variable values correspond to  $i \in I$ . Let  $\gamma = \max_{i \in I} \text{diam}(Y_i)$  and  $\beta = \min_{i \in I} \|G_i^{-1}\|$ , where  $\text{diam}$  indicates the maximum distance between any two points. If  $\Delta_k > \gamma\beta$ , then Lemma 5.1 (with  $v = x_k$ ) ensures that any trial point  $u \in M_k$  either satisfies  $u^c = x_k^c$  or would have lied outside of  $\bigcup_{i \in I} Y_i$ . Then if  $\Delta_k > \gamma\beta$ , no more than  $i_{\max}$  successful iterations will occur before  $\Delta_k$  falls below  $\gamma\beta$ . Thus,  $\Delta_k$  is bounded above by  $\gamma\beta(\tau^{m_{\max}})^{i_{\max}}$ , and the result follows by setting  $r^u$  large enough so that  $\Delta_0\tau^{r^u} \geq \gamma\beta(\tau^{m_{\max}})^{i_{\max}}$ .  $\square$

**Theorem 5.3.** The mesh size parameters satisfy  $\liminf_{k \rightarrow +\infty} \Delta_k = 0$ .

*Proof.* (Torczon [31]) Suppose by way of contradiction that there exists a negative integer  $r^\ell$  such that  $0 < \Delta_0\tau^{r^\ell} \leq \Delta_k$  for all integer  $k \geq 0$ . Combining (3.7) with Lemma 5.2 implies that for any integer  $k \geq 0$ ,  $r_k$  takes its value from among the integers of the finite set  $\{r^\ell, r^\ell + 1, \dots, r^u\}$ . Therefore,  $r_k$  and  $\Delta_k$  can only take a finite number of values for all  $k \geq 0$ .

Since  $x_{k+1} \in M_k$ , (3.2) ensures that  $x_{k+1}^c = x_k^c + \Delta_k D^i z_k$  for some  $z_k \in \mathbb{Z}_+^{|D^i|}$  and  $1 \leq i \leq i_{\max}$ . By repeatedly applying this equation and substituting  $\Delta_k = \Delta_0\tau^{r^k}$ , it follows that, for any integer  $N \geq 1$ ,

$$\begin{aligned} x_N^c &= x_0^c + \sum_{k=1}^{N-1} \Delta_k D^i z_k \\ &= x_0^c + \Delta_0 D^i \sum_{k=1}^{N-1} \tau^{r^k} z_k = x_0^c + \frac{p^{r^\ell}}{q^{r^u}} \Delta_0 D^i \sum_{k=1}^{N-1} p^{r^k - r^\ell} q^{r^u - r^k} z_k, \end{aligned}$$

where  $p$  and  $q$  are relatively prime integers satisfying  $\tau = \frac{p}{q}$ . Since  $p^{r^k - r^\ell} q^{r^u - r^k} z_k$  is an integer for any  $k$ , it follows that the continuous part of all iterates having the same discrete variable values lies on the translated integer lattice generated by  $x_0^c$  and the columns of  $\frac{p^{r^\ell}}{q^{r^u}} \Delta_0 D^i$ . Moreover, the discrete part of all iterates also lies on the integer lattice  $X^d \subset \mathbb{Z}^{n^d}$ .

Therefore, since all iterates belong to a compact set, there must be only a finite number of different iterates, and thus one of them must be visited infinitely many times. Therefore, the mesh coarsening rule in (3.5) is only applied finitely many times, and the mesh refining rule in (3.6) is applied infinitely many times. This contradicts the hypothesis that  $\Delta_0\tau^{r^i}$  is a lower bound for the mesh size parameter.  $\square$

These results show the necessity of forcing the set of directions to satisfy  $D^i = G_i Z_i$ . Under Assumption A1, this ensures that the mesh has only a finite number of points in  $X$ , which means that there can only be a finite number of consecutive unfiltered mesh points. Assumption A2 is included to simply ensure that this construction is maintained in the presence of linear constraints. Audet and Dennis [7] provide an example in which a different construction yields a mesh that is dense in  $X$ . In this case, Lemma 5.1 cannot be satisfied, and convergence of  $\Delta_k$  to zero is not guaranteed. As stated earlier, a sufficient condition for Assumption A2 to hold is that  $G_i = I, i = 1, 2, \dots, i_{\max}$  and that the coefficient matrix  $A$  is rational [24].

We should note also that the rationality of  $\tau$  is essential for convergence. Audet [4] gives an example in which an irrational value for  $\tau$  generates a sequence satisfying  $\liminf_{k \rightarrow +\infty} \Delta_k > 0$ .

### 5.2 Refining Subsequences

Since  $\Delta_k$  shrinks only at iterations in which no mesh isolated filter point is found, Theorem 5.3 guarantees that the Filter-MVPS algorithm has infinitely many such iterations. We are particularly interested in subsequences of iterates that correspond to these points. We now include the following two useful definitions.

**Definition 5.4.** A subsequence of mesh isolated filter points  $\{p_k\}_{k \in K}$  (for some subset of indices  $K$ ) is said to be a *refining subsequence* if  $\{\Delta_k\}_{k \in K}$  converges to zero.

**Definition 5.5.** Let  $\{v_k\}_{k \in K}$  be either a refining subsequence or a corresponding subsequence of extended poll endpoints, and let  $\hat{v}$  be a limit point of the subsequence. A direction  $d \in D$  is said to be a *limit direction* of  $\hat{v}$  if  $v_k + \Delta_k(d, 0)$  belongs to  $X$  and is filtered for infinitely many  $k \in K$ .

In [6], Audet and Dennis prove the existence of a limit point

$$\hat{p} = \lim_{k \in K} p_k, \quad (5.1)$$

where  $K$  is the index set corresponding to a refining subsequence. They also establish two other important classes of limit points,

$$\hat{y} = \lim_{k \in K} y_k \quad \text{and} \quad \hat{z} = (\hat{z}^c, \hat{y}^d) = \lim_{k \in K} z_k, \quad (5.2)$$

where each  $z_k \in X$  is the endpoint of the EXTENDED POLL step initiated at  $y_k \in \mathcal{N}(p_k)$ .

The notation used in (5.1) and (5.2) that describes these specific subsequences and their limit points will be retained and used throughout the remainder of this paper. Many of the theorems that follow will require the assumption that  $\hat{y} \in \mathcal{N}(\hat{p})$ .

### 5.3 Background for Optimality Results

In this subsection, we provide some additional background material, based on the ideas of the Clarke calculus, along with a new definition and theorem that will be used in the convergence theorems. Some of these ideas have been used in proofs by Audet and Dennis [7, 8] in the context of certain limit points of the GPS algorithm, and the new definition allows us to generalize slightly their hypotheses.

First, the following definitions from [13] are needed. They apply to any function  $g : \mathbb{R}^n \rightarrow \mathbb{R}$  that is Lipschitz near a point  $x \in \mathbb{R}^n$ .

- The *generalized directional derivative* of  $g$  at  $x$  in the direction  $v$  is given by

$$g^\circ(x; v) := \limsup_{y \rightarrow x, t \downarrow 0} \frac{g(y + tv) - g(y)}{t},$$

where  $t$  is a positive scalar.

- The *generalized gradient* of  $g$  at  $x$  is the set

$$\partial g(x) := \{\zeta \in \mathbb{R}^n : g^\circ(x; v) \geq v^T \zeta \text{ for all } v \in \mathbb{R}^n\}.$$

- $g$  is *strictly differentiable* at  $x$  if, for all  $v \in \mathbb{R}^n$ ,

$$\lim_{y \rightarrow x, t \downarrow 0} \frac{g(y + tv) - g(y)}{t} = \nabla g(x)^T v.$$

The following is a generalization of the previous definition.

**Definition 5.6.** Let  $X$  be a convex subset of  $\mathbb{R}^n$ . Let  $T_X(x)$  denote the tangent cone to  $X$  at  $x \in X$ . A function  $g$  is said to be *strictly differentiable with respect to  $X$*  at  $x \in X$  if, for all  $v \in T_X(x)$ ,

$$\lim_{y \rightarrow x, y \in X, t \downarrow 0} \frac{g(y + tv) - g(y)}{t} = \nabla g(x)^T v.$$

Theorem 5.7 below essentially establishes first-order necessary conditions for optimality with respect to the continuous variables in a mixed variable domain. The assumptions on  $g$  given here are slightly weaker than the strict differentiability assumption used in [7] to establish first-order results for GPS limit points – but only in the presence of linear constraints. Without linear constraints, Definition 5.6 clearly reduces to that of strict differentiability.

However, we first introduce new notation, so that  $g'(x; (d, 0))$  denotes the directional derivative at  $x$  with respect to the continuous variables in the direction  $d \in \mathbb{R}^{n^c}$  (i.e., while holding the discrete variables constant – hence the  $0 \in \mathbb{Z}^{n^d}$ ),  $g^\circ(x; (d, 0))$  denotes the Clarke generalized directional derivative at  $x$  with respect to the continuous variables, and  $\partial^c g(x)$  represents the generalized gradient of  $g$  at  $x$  with respect to the continuous variables. This convention is used throughout Section 5.

**Theorem 5.7.** Let  $x = (x^c, x^d) \in X \subseteq \mathbb{R}^{n^c} \times \mathbb{Z}^{n^d}$ . Suppose the function  $g$  is strictly differentiable with respect to  $X^c$  at  $x$ . If  $D \in \mathbb{R}^{n^c}$  positively spans the tangent cone  $T_{X^c}(x)$ , and if  $g^\circ(x; (d, 0)) \geq 0$  for all  $d \in D \cap T_{X^c}(x)$ , then  $x$  is a KKT point of  $g$  with respect to the continuous variables. Moreover, if  $X^c = \mathbb{R}^{n^c}$  or if  $x^c$  lies in the interior of  $X^c$ , then  $g$  is strictly differentiable at  $x$  with respect to the continuous variables and  $\nabla^c g(x) = 0$ .

*Proof.* Under the hypotheses given, let  $D$  be a set of vectors that positively spans  $T_{X^c}(x)$ , and let  $v \in T_{X^c}(x)$  be arbitrary. Then  $v = \sum_{i=1}^{|D|} \alpha_i d_i$  for some  $\alpha_i \geq 0$  and  $d_i \in D$ ,  $i = 1, 2, \dots, |D|$ . Then

$$\nabla^c g(x)^T v = \sum_{i=1}^{|D|} \alpha_i \nabla^c g(x)^T d_i = \sum_{i=1}^{|D|} \alpha_i g^\circ(x; (d_i, 0)) \geq 0,$$

since all the terms of the final sum are nonnegative.

If  $X^c = \mathbb{R}^{n^c}$ , or if  $x^c$  lies in the interior of  $X^c$ , then  $T_{X^c}(x) = \mathbb{R}^{n^c}$  and  $g$  is strictly differentiable at  $x$ . Since we have  $\nabla^c g(x)^T v \geq 0$  for all  $v \in \mathbb{R}^{n^c}$ , including  $-v$ , we also have  $\nabla^c g(x)^T v \leq 0$  for all  $v \in \mathbb{R}^{n^c}$ . Therefore,  $0 = \nabla^c g(x)$ . □

**5.4 Results for the Constraint Violation Function**

Section 5.2 establishes the existence of the limit points  $\hat{p}$ ,  $\hat{y}$ , and  $\hat{z}$ , as shown in (5.1) and (5.2). While the next result applies to more general limit points of the algorithm, the remainder of the results in this section apply to these specific limit points. This format will be repeated in Section 5.5 as well. We remind the reader that, since we have chosen  $h$  as specified in (4.1),  $h$  inherits whatever smoothness  $C$  possesses [8]. This enables us to state our the hypotheses in terms of  $C$ , rather than  $h$ , in the results that follow.

The first result, which is similar to a theorem in [7] for  $f$ , requires a very mild condition on  $h$ . Note that this result will not hold for  $f$  without an additional assumption because there is no guarantee that any subsequence of objective function values is nonincreasing.

**Proposition 5.8.** If  $C$  is lower semi-continuous with respect to the continuous variables at a limit point  $\bar{p}$  of poll centers  $\{p_k\}$ , then  $\lim_k h(p_k)$  exists and is greater than or equal to  $h(\bar{p}) \geq 0$ . If  $C$  is continuous at every limit point of  $\{p_k\}$ , then every limit point has the same function value.

*Proof.* If  $C(\bar{p}) \leq 0$ , then  $h(\bar{p}) = 0$ , and the result follows trivially. Now let  $C_i(\bar{p}) > 0$  for some  $i = 1, 2, \dots, p$ , in which case,  $h(\bar{p}) > 0$ . Then  $\bar{p}$  is a limit point of a sequence of least infeasible points  $p_k^I$ , which is monotonically nonincreasing. Since  $C$ , and hence  $h$ , is lower semi-continuous at  $\bar{p}$ , we know that for any subsequence  $\{p_k\}_{k \in K}$  of poll centers that converges to  $\bar{p}$ ,  $\liminf_{k \in K} h(p_k) \geq h(\bar{p}) \geq 0$ . But the subsequence of constraint violation function values at  $p_k^I$  is a subsequence of a nonincreasing sequence. Thus, the entire sequence is also bounded below by  $h(\bar{p})$ , and so it converges.  $\square$

We now characterize the limit points identified in (5.1) and (5.2) with respect to the constraint violation function  $h$ . The following theorem establishes the local optimality of  $h$  at  $\hat{p}$  with respect to its discrete neighbors. The short proof is nearly identical to one in [6].

**Theorem 5.9.** Let  $\hat{p}$  and  $\hat{y} \in \mathcal{N}(\hat{p})$  be limit points as defined by (5.1) and (5.2). If  $C$  is lower semi-continuous at  $\hat{p}$  and upper semi-continuous at  $\hat{y}$  with respect to the continuous variables, then  $h(\hat{p}) \leq h(\hat{y})$ .

*Proof.* Since  $k \in K$  ensures that  $\{p_k\}_{k \in K}$  are mesh isolated poll centers, we have  $h(p_k) \leq h(y_k)$  for all  $k \in K$ . By the assumptions of lower and upper semi-continuity on  $C$ , and hence  $h$ , along with (5.1) and (5.2), we have  $h(\hat{p}) \leq \lim_{k \in K} h(p_k) \leq \lim_{k \in K} h(y_k) = h(\hat{y})$ .  $\square$

The next two results establish a directional optimality condition for  $h$  at  $\hat{p}$  and  $\hat{z}$  with respect to the continuous variables.

**Theorem 5.10.** Let  $\hat{p}$  be a limit point of a refining subsequence. Under Assumptions A1–A2, if  $C$  is Lipschitz near  $\hat{p}$  with respect to the continuous variables, then  $h^\circ(\hat{p}; (d, 0)) \geq 0$  for all limit directions  $d \in D(\hat{p})$  of  $\hat{p}$ .

*Proof.* Let  $\{p_k\}_{k \in K}$  be a refining subsequence with limit point  $\hat{p}$  and let  $d \in D(\hat{p})$  be a refining direction of  $\hat{p}$ . From the definition of the generalized directional derivative, we have

$$h^\circ(\hat{p}; (d, 0)) = \limsup_{y \rightarrow \hat{p}, t \downarrow 0} \frac{h(y + t(d, 0)) - h(y)}{t} \geq \limsup_{k \in K} \frac{h(p_k + \Delta_k(d, 0)) - h(p_k)}{\Delta_k}.$$

Since  $C$  is Lipschitz near  $\hat{p}$ ,  $h$  is also Lipschitz, and hence finite, near  $\hat{p}$ . Since points that are infeasible with respect to  $X$  are not evaluated by the algorithm, the assumption of  $d$  being a limit direction of  $\hat{p}$  ensures that infinitely many right-hand quotients are defined. All of these quotients must be nonnegative, or else the corresponding POLL step would have found an unfiltered point, a contradiction.  $\square$

**Theorem 5.11.** Let  $\hat{p}, \hat{y} \in \mathcal{N}(\hat{p})$ , and  $\hat{z}$  be limit points defined by (5.1) and (5.2), and let  $\xi > 0$  denote a lower bound on the extended poll triggers  $\xi_k^f$  and  $\xi_k^h$  for all  $k$ . Under Assumptions A1–A2, if  $h(\hat{y}) < h(\hat{p}) + \xi$  and  $C$  is Lipschitz near  $\hat{z}$  with respect to the continuous variables, then  $h^\circ(\hat{z}; (d, 0)) \geq 0$  for all limit directions  $d \in D(\hat{z})$  of  $\hat{z}$ .

*Proof.* From the definition of the generalized directional derivative [13], we have

$$h^\circ(\hat{z}; (d, 0)) = \limsup_{y \rightarrow \hat{z}, t \downarrow 0} \frac{h(y + t(d, 0)) - f(y)}{t} \geq \limsup_{k \in K} \frac{h(z_k + \Delta_k(d, 0)) - h(z_k)}{\Delta_k}.$$



Since  $C$  is Lipschitz near  $\hat{z}$ ,  $h$  is also Lipschitz, and hence finite, near  $\hat{z}$ . Since  $h(\hat{y}) < h(\hat{p}) + \xi$  ensures that extended polling was triggered around  $y_k \in \mathcal{N}(p_k)$  for all sufficiently large  $k \in K$ , and since  $d$  is a limit direction of  $\hat{z}$ , it follows that  $z_k + \Delta_k(d, 0) \in X$  infinitely often in  $K$ , and infinitely many of the right-hand quotients are defined. All of these quotients must be nonnegative, since for  $k \in K$ ,  $z_k$  is an extended poll endpoint.  $\square$

For bound or linear constraints, in order to guarantee the existence of limit directions, for which Theorem 5.10 applies, each  $D^i \subset D, i = 1, 2, \dots, i_{\max}$  is constructed in accordance with the algorithm given in [24] to generate a sufficiently rich set of directions to ensure conformity to  $X^c$  (see Definition 3.1).

The next two corollaries establish conditions on  $h$  under which  $\hat{p}$  and  $\hat{z}$  satisfy a first-order optimality condition with respect to the continuous variables.

**Corollary 5.12.** Let  $\hat{p}$  be the limit of a refining subsequence with limit directions  $D(\hat{p})$ , and suppose  $C$  is strictly differentiable with respect to  $X^c$  at  $\hat{p}$ . Then under Assumptions A1–A2,  $\nabla^c h(\hat{p})^T w \geq 0$  for all  $w \in T_{X^c}(\hat{p})$ . Moreover, if  $X^c = \mathbb{R}^{n^c}$ , or if  $\hat{p}$  lies in the interior of  $X^c$ , then  $\nabla^c h(\hat{p}) = 0$ .

*Proof.* Since the rule for selecting  $D_k(p_k)$  conforms to  $X^c$  for some  $\epsilon > 0$ , and since there are finitely many linear constraints,  $D_k(p_k) \rightarrow D(\hat{p})$ , and  $D(\hat{p})$  positively spans  $T_{X^c}(\hat{p})$ . Theorem 5.10 guarantees that  $h^\circ(\hat{p}; (d, 0)) \geq 0$  for all  $d \in D(\hat{p})$ , and since the strict differentiability of  $C$  applies to  $h$ , the result follows directly from Theorem 5.7.  $\square$

**Corollary 5.13.** Let  $\hat{p}, \hat{y} \in \mathcal{N}(\hat{p})$ , and  $\hat{z}$  be limit points defined by (5.1) and (5.2), and let  $\xi > 0$  denote a lower bound on the extended poll triggers  $\xi_k^f$  and  $\xi_k^h$  for all  $k$ . Let  $D(\hat{z})$  denote the limit directions of  $\hat{z}$ , and suppose  $C$  is strictly differentiable with respect to  $X^c$  at  $\hat{z}$ . If  $h(\hat{y}) < h(\hat{p}) + \xi$ , then under Assumptions A1–A2,  $\nabla^c h(\hat{z})^T w \geq 0$  for all  $w \in T_{X^c}(\hat{z})$ . Furthermore, if  $X^c = \mathbb{R}^{n^c}$  or  $\hat{z}$  lies in the interior of  $X^c$ , then  $\nabla^c h(\hat{z}) = 0$ .

*Proof.* Since the rule for selecting  $D_k(z_k)$  conforms to  $X^c$  for some  $\epsilon > 0$ , and since there are finitely many linear constraints,  $D_k(z_k) \rightarrow D(\hat{z})$ , and  $D(\hat{z})$  positively spans  $T_{X^c}(\hat{z})$ . Theorem 5.11 ensures that  $h^\circ(\hat{z}; (d, 0)) \geq 0$  for all  $d \in D(\hat{z})$ , and since the strict differentiability of  $C$  applies to  $h$ , the result follows directly from Theorem 5.7.  $\square$

As a consequence of these two corollaries and Theorem 5.9, we have shown that, under the assumptions given,

- $\hat{p}$  is a KKT point for  $h$  with respect to the continuous variables.
- $h(\hat{p}) \leq h(\hat{y})$  for all  $\hat{y} \in \mathcal{N}(\hat{p})$ .
- For any  $\hat{y} \in \mathcal{N}(\hat{p})$  satisfying  $h(\hat{p}) = h(\hat{y})$ , the extended poll endpoint  $\hat{z}$  associated with  $\hat{y}$  is a KKT point with respect to the continuous variables.

In the third condition, we actually have  $h(\hat{p}) = h(\hat{y}) = h(\hat{z})$ . Naturally, we would prefer  $\hat{y}$  to also be a KKT point, but this requires a stronger hypothesis on the algorithm, which is shown in [6]. If the three limit points are all infeasible, then it is probably the case that  $\hat{y} = \hat{z}$ . If the limit points are feasible, then this issue is moot, since we achieve the desired condition that  $\hat{p}$  minimizes  $h$ .

Note also that, although there is no guarantee of convergence to a feasible point, these results (with respect to the continuous variables) are no weaker than those obtained for

many other optimization algorithms. In fact, the typical assumption for SQP methods is that the Jacobian of the active constraints is full rank everywhere. This is a very strong assumption and easily implies that every local minimizer of  $h$  is a feasible point.

### 5.5 Results for the Objective Function

We now address the properties of certain limit points with respect to the objective function  $f$ . Unfortunately, in order to obtain results for  $f$  that are similar to those for  $h$ , an additional hypothesis must be added; namely, that the best feasible point must be chosen as the poll center for all but a finite number of iterations. Furthermore, convergence to a KKT point (with respect to the continuous variables) cannot be guaranteed, but we will show a similar result to that of [8], in which a cone is identified whose polar contains the negative gradient.

The first result, under very mild conditions, is similar to Proposition 5.8.

**Proposition 5.14.** Under Assumption A1, there exists at least one limit point  $\bar{p}$  of the iteration sequence  $\{p_k\}$  of poll centers. Suppose that this sequence was generated by polling around the best feasible point for all but a finite number of iterations. If  $f$  is lower semi-continuous at  $\bar{p}$  with respect to the continuous variables and  $h$  is continuous at  $\bar{p}$  with respect to the continuous variables, then  $\lim_k f(p_k)$  exists and is greater than or equal to  $f(\bar{p})$ , which is finite. If  $f$  is continuous at every limit point of  $\{p_k\}$ , then every limit point has the same function value.

*Proof.* First,  $p_k = p_k^F$ , hence  $h(p_k) = 0$ , for all but finitely many  $k$ . Thus,  $f(p_k)$  is nonincreasing for all sufficiently large  $k \in K$ . Since  $f$  is lower semi-continuous at  $\bar{p}$ , we know that for any subsequence  $\{p_k\}_{k \in K}$  of poll centers converging to  $\bar{p}$ ,  $\liminf_{k \in K} f(p_k) \geq f(\bar{p})$ . But the subsequence of function values is a subsequence of a nonincreasing sequence (for sufficiently large  $k$ ). Thus, for sufficiently large  $k$ , the sequence is also bounded below by  $f(\bar{p})$ , and so it converges.  $\square$

The remainder of this section contains results for the limit points described by (5.1) and (5.2). The following result, which is similar to Theorem 5.9, establishes optimality conditions with respect to the discrete set of neighbors.

**Theorem 5.15.** Let  $\hat{p}$  be a limit point of a refining subsequence  $\{p_k\}_{k \in K}$  in which  $p_k$  was chosen as the best feasible point for all but a finite number of iterations, and let  $\hat{y} \in \mathcal{N}(\hat{p})$  be defined by (5.2). If  $f$  is lower semi-continuous at  $\hat{p}$  and upper semi-continuous at  $\hat{y}$  with respect to the continuous variables, then  $f(\hat{p}) \leq f(\hat{y})$ .

*Proof.* Since  $p_k = p_k^F$  for all but a finite number of iterations,  $h(p_k) = 0$ , and hence  $f(p_k) \leq f(y_k)$ , for all sufficiently large  $k \in K$ . Taking limits and applying (5.1) and 5.2, the assumptions of lower and upper semi-continuity of  $f$  at  $\hat{p}$  and  $\hat{y}$ , respectively, yields the result:  $f(\hat{p}) \leq \lim_{k \in K} f(p_k) \leq \lim_{k \in K} f(y_k) = f(\hat{y})$ .  $\square$

The next two results establish conditions under which Clarke generalized directional derivatives are nonnegative at  $\hat{p}$  and  $\hat{z}$ , respectively.

**Theorem 5.16.** Let  $\hat{p}$  be a limit point of a refining subsequence  $\{p_k\}_{k \in K}$  in which  $p_k$  was chosen as the best feasible point for all but a finite number of iterations, and let  $d \in D$  be a limit direction of  $\hat{p}$ . Under Assumptions A1–A2, if  $f$  is Lipschitz near  $\hat{p}$  with respect to the continuous variables, then  $f^\circ(\hat{p}; (d, 0)) \geq 0$ .

*Proof.* Since  $p_k = p_k^F$  for all but a finite number of iterations, it follows that  $h(p_k) = 0$  for sufficiently large  $k \in K$ . From the definition of the generalized directional derivative [13], we have that

$$f^\circ(\hat{p}; (d, 0)) = \limsup_{y \rightarrow \hat{p}, t \downarrow 0} \frac{f(y + t(d, 0)) - f(y)}{t} \geq \limsup_{k \in K} \frac{f(p_k + \Delta_k(d, 0)) - f(p_k)}{\Delta_k},$$

which is nonnegative, since the terms in the right-hand quotient are nonnegative for all sufficiently large  $k \in K$ . □

**Theorem 5.17.** Let  $\hat{p}$  be a limit point of a refining subsequence, and let  $\hat{y} \in \mathcal{N}(\hat{p})$  and  $\hat{z}$  be limit points as defined by (5.2), in which the extended poll endpoints  $z_k$  were chosen as the best feasible point with respect to the local filter  $\mathcal{F}_L$  at all but a finite number of iterations. Let  $d \in D$  be a limit direction for  $\hat{z}$ . Suppose that  $f(\hat{y}) < f(\hat{p}) + \xi$ , where  $\xi > 0$  is a lower bound on the extended poll triggers  $\xi_k^f$  and  $\xi_k^h$  for all  $k$ . Under Assumptions A1–A2, if  $f$  is Lipschitz near  $\hat{z}$  with respect to the continuous variables, then  $f^\circ(\hat{z}; (d, 0)) \geq 0$ .

*Proof.* Since  $z_k = z_k^F$  for all but a finite number of iterations, it follows that  $h(z_k) = 0$  for all sufficiently large  $k \in K$ . From the definition of the generalized directional derivative [13], we have that

$$f^\circ(\hat{z}; (d, 0)) = \limsup_{y \rightarrow \hat{z}, t \downarrow 0} \frac{f(y + t(d, 0)) - f(y)}{t} \geq \limsup_{k \in K} \frac{f(z_k + \Delta_k(d, 0)) - f(z_k)}{\Delta_k},$$

which is nonnegative, since the terms in the right-hand quotient are nonnegative for all sufficiently large  $k \in K$ . □

The next two results describe the optimality conditions for  $f$  at  $\hat{p}$  and  $\hat{z}$  under the assumptions of strict differentiability with respect to  $X^c$ .

As is the case with the Filter GPS algorithm [8], convergence to a KKT point cannot be guaranteed with respect to the continuous domain, since there is no guarantee that the negative gradient lies inside the normal cone; however, we specify a cone, whose polar contains the negative gradient.

**Corollary 5.18.** Let  $\hat{p}$  be a limit point of a refining subsequence  $\{p_k\}_{k \in K}$  in which  $p_k$  was chosen as the best feasible point for all but a finite number of iterations, and let  $V_d$  be the cone generated by all limit directions  $d \in D$  of  $\hat{p}$ . Under Assumptions A1–A2, if  $f$  is strictly differentiable with respect to  $X^c$  at  $\hat{p}$ , then  $-\nabla^c f(\hat{p})$  belongs to the polar  $V_d^\circ$  of  $V_d$ .

*Proof.* By Theorem 5.16,  $f^\circ(\hat{p}; (d, 0)) \geq 0$  for all  $d \in V_d$ , and by Theorem 5.7, we have  $\nabla^c f(\hat{p})^T w \geq 0$  for all  $w \in V_d$ . The result follows from the definition of a polar cone:  $-\nabla^c f(\hat{p}) \in \{v \in \mathbb{R}^n : v^T w \leq 0 \ \forall w \in V_d\}$ . □

**Corollary 5.19.** Let  $\hat{p}$  be a limit point of a refining subsequence, and let  $\hat{y} \in \mathcal{N}(\hat{p})$  and  $\hat{z}$  be limit points as defined by (5.2) in which the extended poll endpoints  $z_k$  were chosen as the best feasible point with respect to the local filter  $\mathcal{F}_L(y_k)$  at all but a finite number of iterations. Suppose that  $f(\hat{y}) < f(\hat{p}) + \xi$ , where  $\xi > 0$  is a lower bound on the extended poll triggers  $\xi_k^f$  and  $\xi_k^h$  for all  $k$ . Let  $V_d$  be the cone generated by all limit directions  $d \in D$  of  $\hat{z}$ . Under Assumptions A1–A2, if  $f$  is strictly differentiable with respect to  $X^c$  at  $\hat{p}$ , then  $-\nabla^c f(\hat{z})$  belongs to the polar  $V_d^\circ$  of  $V_d$ .

*Proof.* By Theorem 5.17,  $f^\circ(\hat{z};(d,0)) \geq 0$  for all  $d \in V_d$ , and by Theorem 5.7, we have  $\nabla^c f(\hat{z})^T w \geq 0$  for all  $w \in V_d$ . The result follows from the definition of a polar cone:  $-\nabla^c f(\hat{z}) \in \{v \in \mathbb{R}^n : v^T w \leq 0 \forall w \in V_d\}$ .  $\square$

The theorems presented in this section are more restrictive than the corresponding theorems in Section 5.4, in that they only apply to the subclass of possible limit points of refining subsequences, in which the best feasible point was chosen as the poll center for all but a finite number of iterations. However, there are more subsequences for which these theorems hold. The results from these three theorems can be satisfied for a larger class of refining subsequences by replacing the assumption of polling around the best feasible point for all but finitely many iterations with the condition that poll points be filtered by the best feasible or least infeasible point, rather than by a different filter point, for all but a finite number of iterations. In particular, Theorem 5.15 holds if we require  $f(p_k) \leq f(y_k)$  for all sufficiently large  $k \in K$ , Theorem 5.16 and Corollary 5.18 hold if we require  $f(p_k) \leq f(p_k + \Delta_k d)$  for all sufficiently large  $k \in K$ , and Theorem 5.17 and Corollary 5.19 hold if we require  $f(z_k) \leq f(z_k + \Delta_k d)$  for all sufficiently large  $k \in K$ . While these conditions are more general than the condition they would replace, they are dependent on the performance of the algorithm and thus cannot be ensured in practice. On the other hand, the choice to poll around the best feasible point for all but a finite number of iterations is easy to implement.

A second point is that these theorems assume that we have feasible points that we can poll around. If this is not the case, then the results of Section 5.4 still hold for  $h$ , and it is still possible, though not guaranteed, that the results of this section can yet hold.

Finally, we point out one other key result that we adapt from [8].

**Proposition 5.20.** *If  $h$  and  $f$  are strictly differentiable with respect to  $X^c$  at poll center  $p_k$ , and if  $\nabla^c f(p_k) \neq 0$ , then there cannot be infinitely many consecutive iterations where  $p_k$  is a mesh isolated poll center.*

*Proof.* Under the assumptions, suppose that there are infinitely many consecutive iterations in which  $p_k$  is a mesh isolated filter point. Let  $d$  be a direction associated with the (constant) subsequence of poll centers such that  $\nabla^c f(p_k)^T d < 0$ .

Since  $h$  is strictly differentiable at  $p_k$  with respect to the continuous variables, there exists an  $\epsilon > 0$  such that either  $h(p_k + \Delta(d,0)) \leq h(p_k) < h_{\max}$ , or  $h(p_k + \Delta(d,0)) > h(p_k)$ , for all  $0 < \Delta < \epsilon$ .

If the first condition is satisfied, then for  $\Delta_k < \epsilon$ , the POLL step will find an unfiltered point, a contradiction. If the second condition is satisfied, then let  $\tilde{h}$  be the smallest value of

$$\{h(x) : h(x) > h(p_k), x \in \mathcal{F}_k\} \cup \{h_{\max}\},$$

and let  $\tilde{f}$  be the corresponding objective function value; i.e., either  $\tilde{f} = f(\tilde{x})$  for the vector  $\tilde{x} \in \mathcal{F}_k$  that satisfies  $h(\tilde{x}) = \tilde{h}$ , or  $\tilde{f} = -\infty$  in the case where  $\tilde{h} = h_{\max}$ . It follows that  $\tilde{h} > h(p_k)$  and  $\tilde{f} < f(p_k)$ . Therefore, for sufficiently small  $\Delta_k < \epsilon$ , we have  $h(p_k) < h(p_k + \Delta_k d) < \tilde{h}$  and  $\tilde{f} < f(p_k + \Delta_k d) < f(p_k)$ ; thus, the trial mesh point is unfiltered, a contradiction.  $\square$

A limitation of this result is that, while it prevents a non-stationary  $p_k$  from being a mesh isolated poll center for infinitely many consecutive iterations, it does not completely prevent the algorithm from stalling there. The algorithm could still generate an infinite number of consecutive iterations in which  $p_k$  is either a mesh isolated filter point or a filter

point that does not generate a new poll center. If, for example,  $p_k$  simply alternates between these two possibilities, then Proposition 5.20 holds, but the algorithm still stalls at  $p_k$ .

As in previous results, the additional hypothesis of  $p_k = p_k^F$  for infinitely many  $k \in K$  would fully prevent stalling because it would force  $h(p_k) = 0$  for infinitely many  $k \in K$ , and the strict differentiability of  $f$  at  $p_k$  means that  $\nabla^c f(p_k)^T d < 0$  for some direction  $d \in D_k(p_k)$ . Thus, for sufficiently large  $k \in K$ ,  $\Delta_k$  is sufficiently small to force  $f(p_k + \Delta_k d) < f(p_k)$ , and the algorithm moves to a new point.

**Remark 5.21.** Many of the results in this and the previous subsections also apply to additional directions, which are specifically identified in [8]. We have not included this in our presentation because it would require an extraordinary amount of additional material to explain it properly. Since it is not possible to ensure convergence to a KKT point, the extra material adds little to the overall convergence theory. Instead, we refer the interested reader to [8] for a thorough discussion.

**5.6 Conclusions**

In summary, if the Assumptions A1–A2 are satisfied, then we get the following hierarchy of convergence results for the Filter-MGPS Algorithm. These results unify the convergence theory of the previous work in pattern search methods. In what follows,  $\bar{p}$  denotes any limit point of the entire GPS iteration sequence,  $p_k^F$  denotes the best feasible point at iteration  $k$ , and  $\hat{p}$ ,  $\hat{y}$ , and  $\hat{z}$  are limit points defined by (5.1) and (5.2).

- If  $C$  is lower semi-continuous at any limit point  $\bar{p}$  of the GPS iteration sequence with respect to the continuous variables, then  $h(\bar{p}) \leq \lim_k h(p_k)$ . A similar result for  $f$  requires that  $p_k = p_k^F$  for all but finitely many  $k$ .
- Every limit point of the iteration sequence at which  $C$  is continuous has the same constraint violation function value  $\lim_k h(p_k)$ , whether or not it is a stationary point for  $h$ . A similar result for  $f$  requires that  $p_k = p_k^F$  for all but finitely many  $k$ .
- If  $C$  is lower semi-continuous at  $\hat{p}$  and upper semi-continuous at  $\hat{y} \in \mathcal{N}(\hat{p})$  with respect to the continuous variables, then  $h(\hat{p}) \leq h(\hat{y})$ . A similar result for  $f$  requires the choice of  $p_k = p_k^F$  for all but finitely many  $k$ .
- If the function  $C$  is Lipschitz near  $\hat{p}$  with respect to the continuous variables, then  $h^\circ(\hat{p}; (d, 0)) \geq 0$  for any limit direction  $d$  of  $\hat{p}$ . A similar result holds at  $\hat{z}$ . Similar results for  $f$  at  $\hat{p}$  and  $\hat{z}$  require the choice of  $p_k = p_k^F$  (or  $z_k = z_k^F$  with respect to the local filter) for all but finitely many  $k$ .
- If  $C$  is strictly differentiable (or its equivalent on the tangent cone) at  $\hat{p}$  with respect to the continuous variables, then  $\hat{p}$  is a first-order stationary point for  $h$  with respect to the continuous variables. A similar result holds at  $\hat{z}$ .
- If  $f$  is strictly differentiable (with respect to the continuous variables) at a limit point  $\hat{p}$  for which the choice of  $p_k = p_k^F$  was made for all but finitely many  $k$ , then the polar of the cone formed by the limit directions of  $\hat{p}$  contains  $-\nabla f(\hat{p})$ . While a first order stationary cannot be guaranteed in general, it can be if  $C(\hat{p}) < 0$ . A similar result holds at  $\hat{z}$ .

It is unfortunate that convergence of the algorithm to a first-order stationary point (with respect to the continuous variables) cannot be guaranteed. In order for this to occur,

we would need  $-\nabla f(\hat{p})$  to be contained in the normal cone. Instead, it is contained in a different cone generated by certain limit directions, which is all we can show for this class of algorithms. Nevertheless, its effectiveness in practice has been demonstrated [2]. At some point in the future, we hope to apply the new class of mesh-adaptive direct search (MADS) algorithms [9] to mixed variable problems, since it is a generalization of pattern search that possesses stronger convergence properties [9, 3] for problems with nonlinear constraints.

## References

- [1] M.A. Abramson, Pattern search algorithms for mixed variable general constrained optimization problems, Department of Computational and Applied Mathematics, Rice University, August, 2002.
- [2] M.A. Abramson, Mixed variable optimization of a load-bearing thermal insulation system using a filter pattern search algorithm, *Optim. Eng.* 5 (2004) 157–177.
- [3] M.A. Abramson and C. Audet, Second-order convergence of mesh adaptive direct search, *SIAM J. Optim.* 17 (2006) 606–619.
- [4] C. Audet, Convergence results for pattern search algorithms are tight, *Optim. Eng.* 5 (2004) 101–122.
- [5] C. Audet, A.J. Booker, J.E. Dennis, Jr., P.D. Frank and D.W. Moore, A surrogate-model-based method for constrained optimization, AIAA Paper 2000–4891, Presented at the 8th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, California, 2000.
- [6] C. Audet and J.E. Dennis, Jr., Pattern search algorithms for mixed variable programming, *SIAM J. Optim.* 11 (2000) 573–594.
- [7] C. Audet and J.E. Dennis, Jr., Analysis of generalized pattern searches, *SIAM J. Optim.* 13 (2003) 889–903.
- [8] C. Audet and J.E. Dennis, Jr., A pattern search filter method for nonlinear programming without derivatives, *SIAM J. Optim.* 14 (2004) 980–1010.
- [9] C. Audet and J.E. Dennis, Jr., Mesh adaptive direct search algorithms for constrained optimization *SIAM J. Optim.* 17 (2006) 188–217.
- [10] A.J. Booker, J.E. Dennis, Jr., P.D. Frank, D.W. Moore and D. B. Serafini, Managing surrogate objectives to optimize a helicopter rotor design—further experiments, AIAA Paper 1998–4717, Presented at the 8th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, 1998.
- [11] A.J. Booker, J.E. Dennis, Jr., P.D. Frank, D.B. Serafini and V. Torczon, Optimization using surrogate objectives on a helicopter test example, in *Optimal Design and Control*, Progress in Systems and Control Theory, J. Borggaard, J. Burns, E. Cliff and S. Schreck (eds.), Birkhäuser, Cambridge, Massachusetts, 1998, pp. 49–58.
- [12] A.J. Booker, J.E. Dennis, Jr., P.D. Frank, D.B. Serafini, V. Torczon and M.W. Trosset, A rigorous framework for optimization of expensive functions by surrogates, *Struct. Optim.* 17 (1999) 1–13.

- [13] F.H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983. Reissued in 1990 by SIAM Publications, Philadelphia, as Vol. 5 in the series, Classics in Applied Mathematics.
- [14] A.R. Conn, N.I.M. Gould and P.L. Toint, A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds, *SIAM J. Numer. Anal.* 28 (1991) 545–572.
- [15] C. Davis, Theory of positive linear dependence, *Amer. J. Math.* 76 (1954) 733–746.
- [16] R. Fletcher, N.I.M. Gould, S. Leyffer, P.L. Toint and A. Wächter, On the global convergence of trust-region SQP-filter algorithms for general nonlinear programming, *SIAM J. Optim.* 13 (2002) 635–659.
- [17] R. Fletcher and S. Leyffer, Nonlinear programming without a penalty function, *Math. Program. Series A*, 91 (2002) 239–269.
- [18] R. Fletcher, S. Leyffer and P.L. Toint, On the global convergence of an SLP-filter algorithm, Dundee University, Department of Mathematics, 1998.
- [19] R. Fletcher, S. Leyffer and P.L. Toint, On the global convergence of a filter-SQP algorithm, *SIAM J. Optim.*, 13 (2002) 44–59.
- [20] F. John, Extremum problems with inequalities as subsidiary conditions, in *Studies and Essays: Courant Anniversary Volume*, K.O. Friedrichs and O.E. Neugebauer and J.J. Stoker (eds.), Wiley-Interscience, New York, 1948, pp. 187–204.
- [21] M. Kokkolaras, C. Audet and J.E. Dennis, Jr., Mixed variable optimization of the number and composition of heat intercepts in a thermal insulation system, *Optim. Eng.* 2 (2001) 5–29.
- [22] R. M. Lewis and V. Torczon, Rank ordering and positive bases in pattern search algorithms, Technical Report 96–71, Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, Virginia 23681–2199, 1996.
- [23] R.M. Lewis and V. Torczon, Pattern search algorithms for bound constrained minimization, *SIAM J. Optim.* 9 (1999) 1082–1099.
- [24] R.M. Lewis and V. Torczon, Pattern search methods for linearly constrained minimization, *SIAM J. Optim.* 10 (2000) 917–941.
- [25] R.M. Lewis and V. Torczon, A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds, *SIAM J. Optim.* 12 (2002) 1075–1089.
- [26] S. Lucidi and V. Piccialli, A derivative-based algorithm for a particular class of mixed variable optimization problems, *Optim. Methods Softw.* 17 (2004) 317–387.
- [27] S. Lucidi, V. Piccialli and M. Sciandrone, An algorithm model for mixed variable programming *SIAM J. Optim.* 15 (2005) 1057–1084.
- [28] A.L. Marsden, M. Wang, J.E. Dennis, Jr. and P. Moin, Optimal aeroacoustic shape design using the surrogate management framework., *Optim. Eng.* 5 (2004) 235–262.

- [29] K.R. O'Reilly, Quantitative object reconstruction using Abel transform tomography and mixed variable optimization, Graduate School of Engineering and Management, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, 2006.
- [30] T.A. Sriver, J.W. Chrissis and M.A. Abramson, Pattern search ranking and selection algorithms for mixed variable simulation-based optimization, 2004, preprint.
- [31] V. Torczon, On the convergence of pattern search algorithms, *SIAM J. Optim.* 7 (1997) 1–25.

---

*Manuscript received 6 May 2006*  
*revised 1 November 2006*  
*accepted for publication 8 January 2007*

MARK A. ABRAMSON  
Air Force Institute of Technology  
Department of Mathematics and Statistics  
2950 Hobson Way, Building 641  
Wright Patterson AFB, Ohio, 45433 USA  
E-mail address: `Mark.Abramson@afit.edu`

CHARLES AUDET  
École Polytechnique de Montréal and GERAD  
Département de Mathématiques et de Génie Industriel  
C.P. 6079, Succ. Centre-ville  
Montréal (Québec), H3C 3A7 Canada  
E-mail address: `Charles.Audet@gerad.ca`

J.E. DENNIS JR.  
Rice University, Department of Computational and Applied Mathematics  
8419 42nd Ave SW, Seattle, Washington, 98136 USA  
E-mail address: `dennis@caam.rice.edu`