Analysis of Generalized Pattern Searches

Charles Audet and J. E. Dennis, Jr.

CRPC-TR00814
February 2000

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

Submitted February 2000

Analysis of Generalized Pattern Searches

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

February 4, 2000

Abstract: This paper contains a new analysis for the Generalized Pattern Search (GPS) methods of Torczon and Lewis and Torczon. The two novel aspects are that the proofs are much shorter, and they use weaker continuity assumptions. Specifically, under very mild conditions, the method finds an interesting limit point even if the objective function is not continuous and is even extended valued. If the objective is Lipschitz near the limit point, then appropriate directional derivatives of the objective are zero. If the objective is strictly differentiable at the limit point, then the gradient exists and is zero. The results here show the power of GPS on some classes of real problems better than the previous analysis for continuously differentiable objectives.

Key words: Pattern search algorithm, linearly constrained optimization, surrogate-based optimization, nonsmooth optimization, derivative-free convergence analysis.

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

1 Introduction

GPS is a flexible class of generalized pattern search algorithms defined by Torczon [15] for derivative-free unconstrained optimization. Lewis and Torczon extended the GPS framework to bound constrained optimization [13] and more generally for problems with a finite number of linear constraints [14]. Our purpose here is to provide an alternative to their analysis for all these cases by providing a simple analysis for the last case.

This new analysis is more satisfying mathematically, and it shows more clearly the power of the methods. For example, these methods find their greatest utility in problems where smoothness properties of the objective are problematic. Here, we show the existence of an interesting limit point for any GPS iteration without assuming that the function is continuous or finite valued. Then we add progressively stronger smoothness assumptions to obtain correspondingly stronger results. We obtain the same conclusions for the same algorithms as Torczon and Lewis-Torczon without assuming more than local smoothness at the limit.

The optimization problem considered in this paper is:

$$\min_{x \in \Omega} f(x)$$

but for convergence, we assume only that the objective function $f : \Re^n \to \Re \cup \{\infty\}$. In fact, this generality is not merely a mathematical conceit. For many practical problems, a call to the subroutine that evaluates $f(x)$ may result in no value being returned, which we model as $f(x) = \infty$. This issue is discussed in detail and the effectiveness of GPS for such problems is illustrated in [3]. The issue is discussed as well in [5], [6]. This paper provides the analysis to back up the observation, given for example in Hough, Kolda and Torczon [12], that "even if the theory for GPS requires continuous differentiability of the objective function, these methods can be effective on nondifferentiable problems since they do not rely explicitly on derivatives."

The way of handling constraints here, and indeed the entire algorithm, will be the same as in [13] and [14]. Specifically, we will apply the algorithm not really to f, but to the function $f_\Omega = f + \psi_\Omega$, where ψ_Ω is the indicator function for Ω. It is zero on Ω and ∞ elsewhere. We will assume as in [14] that Ω is the feasible region defined by a finite set of linear constraints: $\Omega = \{x \in \Re^n : \ell \leq Ax \leq u\}$ where $A \in \mathcal{Q}^{m \times m}$, $\ell, u \in \Re^m \cup \{\pm\infty\}$ and $\ell < u$. We can prove some results in a more general context, but we will treat those as asides when we think they are appropriate.

The point is that to get the really strong conclusions requires conditions on the conformity of the geometry of the algorithm with the geometry of Ω. Sticking with Ω considered by Lewis and Torczon will allow us to invoke a version of the conditions they worked out in a very satisfying way for a finite number of linear constraints. These conditions are unlikely to be realized for more general constraints, the key restriction being that there must be a single finite set of generators for all the tangent cones to the boundary of the feasible region. Still, understanding linear constraints is a first step, as is understanding these are probably the most general constraints for which the appealing f_Ω approach is effective.

The remainder of the paper is as follows: in the next section, we will give a brief description of the GPS algorithm class. We adhere to a slightly different, but equivalent version of the Lewis and Torczon algorithm, because our major interest in these algorithms is for problems where they are used with inexpensive surrogates for the expensive objectives they are to be applied to. To see how this is organized, see [3] and [4]. In Section 3, we give the key result and an easy corollary for unconstrained problems before we go on to results for the Lewis and Torczon algorithm. We end with some acknowledgments and conclusions.

2 Generalized Pattern Search Algorithms

Generalized pattern search algorithms for unconstrained or simply constrained minimization generate a sequence of iterates $\{x_k\}$ in $\Re^n$ with non-increasing objective function values. Because of our interest in surrogate-based optimization, we like to view each iteration as being divided into a SEARCH and a POLL phase. For each SEARCH, the objective function is evaluated at a finite number of points on a mesh (a discrete subset of $\Re^n$ defined below) to try to find one that yields a lower objective function value than the incumbent. Any strategy may be used to select the mesh points that are to be candidates for the next iteration, as long as only a finite number of points (including none) are selected. Before declaring the iteration unsuccessful, refining the mesh, and setting $x_{k+1} = x_k$, it is required that the neighboring mesh points be POLLED to see if any one yields a lower function value. Only after a failed poll of the neighbors can an iteration be declared unsuccessful.

If the iteration is successful, then the new point $x_{k+1} \neq x_k$ has a strictly lower objective function value, the mesh size parameter is kept the same or increased, and the process is reiterated. Indeed, as long as the SEARCH steps are succeeding, one would likely choose trial points on a coarser submesh than the current mesh. Our experience with surrogate-based SEARCH steps [3], [4] is that a great deal of progress can be made with few function values, and $\mathcal{O}(n)$ function values are needed only for unsuccessful POLL steps, which indicate that the mesh needs to be refined.

Pattern search algorithms are defined through a finite set of matrices $\mathcal{S}$, each of whose columns are a positive spanning set in $\Re^n$, i.e., the nonnegative linear combinations of the columns of any such set S in $\mathcal{S}$ span $\Re^n$. Moreover, for technical convergence reasons (Torczon's [15] proof of Theorem 3.2) every column s of each matrix must be generated from a single matrix $G^B \in \Re^{n\times n}$ and from a finite set of integer generating matrices $G_\ell \in \mathcal{Z}^{n\times n}$ for $\ell = 1, 2, \ldots, \ell_{max}$ as follows: $s = G^B G_\ell z$ for some $z \in \mathcal{Z}^n$. The current mesh M_k is defined through the lattices spanned by the elements of $\mathcal{S}$: $M_k = \{x_k + \Delta_k S z : z \in \mathcal{Z}^{n_S}, S \in \mathcal{S}\}$, where $\Delta_k > 0$ is the mesh size parameter, and n_S is the number of columns of the matrix S.

Before declaring an iteration unsuccessful, the objective function must be tested at the mesh points that neighbor x_k, the current iterate. This defines the poll set $\{x_k + \Delta_k s : s \text{ is a column of } S_k\}$ for some positive spanning matrix $S_k \in \mathcal{S}$. The points of the poll set are thus the neighbors of x_k on its current mesh with respect to the spanning set S_k. If the iteration is unsuccessful, then the mesh is refined. More precisely, Δ_{k+1} is set to $\tau^{m_k^-}\Delta_k$ for $0 < \tau^{m_k^-} < 1$ where $\tau > 1$ is a rational number that remains constant over all iterations and $m_k^- \leq -1$ is an integer bounded below by $-m_{max} \leq 0$.

If the iteration is successful, then one may choose to coarsen the search to carry out far reaching and inexpensive SEARCH steps. In this case, one searches on a submesh coarsened by the rule $\tau^{m_k^+}\Delta_k$ for some $\tau^{m_k^+} \geq 1$ where $m_k^+ \geq 0$ is an integer bounded above by $m_{max} \geq 0$. By modifying the mesh size parameters this way, it follows that for any $k \geq 0$, there exists an integer $r_k \in \mathcal{Z}$ such that $\Delta_k = \tau^{r_k}\Delta_0$, and the next iterate x_{k+1} can always be written as $x_0 + \sum_{i=1}^{k} \Delta_i S_i z_i$ for some z_i in $\mathcal{Z}^{n_{S_i}}$. This observation, together with the definition of the positive spanning sets through G^B and G_ℓ, are essential to the proof of Theorem 3.2.

A Basic GPS Algorithm

- INITIALIZATION:
  Let $x_0 \in \Omega$ be such that $f(x_0)$ is finite, and let M_0 be a mesh on $\Re^n$ defined by $\Delta_0 > 0$ and x_0. Set the iteration counter k to 0.

- SEARCH AND POLL STEPS:
  Perform the SEARCH and possibly the POLL steps (or only part of the steps) until a trial point x_{k+1} with a lower objective function value is found, or when it is shown that no such trial point exist.

  - SEARCH STEP: Evaluate the objective function on a finite subset of feasible trial points on the mesh M_k (the strategy that gives the set of points is usually provided by the user).

  - POLL STEP: Evaluate the objective function on the poll set around x_k defined by S_k and Δ_k.

- PARAMETER UPDATE:
  If the SEARCH or the POLL step produced a feasible iterate $x_{k+1} \in M_k \cap \Omega$ for which $f(x_{k+1}) < f(x_k)$, then declare the iteration *successful* and update $\Delta_{k+1} \geq \Delta_k$.
  Otherwise, set $x_{k+1} = x_k$, declare the iteration *unsuccessful* and update $\Delta_{k+1} < \Delta_k$.
  Increase $k \leftarrow k + 1$ and go back to the SEARCH and POLL step. ■

The SEARCH strategy is the key to effectiveness. The POLL step, as we will see, guarantees some minimizer necessary conditions at least.

3 Key Convergence Results

The main result in this section uses no special assumptions about the feasible regionΩ. This confirms our claim about the ability of the algorithm to do as much as one could reasonably expect for quite general problems (see also the remark following Theorem 3.7). To illustrate the power of this result, we obtain as an immediate corollary (Theorem 3.5) the strongest result yet for the unconstrained case.

Our convergence analysis of GPS algorithm is based on the standard (see [2], [8], [9], [10] and [11]) assumption that all iterates produced by the algorithm lie in a compact set. A nice sufficient condition for this to hold for pattern search methods is given in [14]: the level set $\{x \in \Omega : f(x) \leq f(x_0)\}$ is compact. We can not assume that the set is compact because we allow discontinuities and even $f(x) = \infty$ sometimes, and so we do not know that the set is closed. However we can assume that the set is precompact or bounded.

Whatever we assume to ensure that the iterates are in a compact set, this already means that there are convergent subsequences of the iteration sequence, but we will identify an interesting set of subsequences using the behavior of the algorithm. Specifically, we will be concerned here with the iterates x_k about which unsuccessful poll steps were conducted. It is only when an iteration is unsuccessful that Δ_k is reduced. This is not to say that other subsequences may not exhibit interesting behavior, but we can prove that these do.

Our first result is that there is a subsequence of such iterations for which the mesh size parameter goes to zero. In order to prove it we require the following lemma. Neither proof depends at all on the smoothness of the objective, rather they use just the definition of the algorithm and rationality of the polling sets.

Lemma 3.1 *The mesh size parameters Δ_k are bounded above by a positive constant independent of the iteration number k.*

Proof. Let $\mathcal{X}$ be a compact set in $\Re^n$ that contains all the iterates. Choose Δ in $\Re$ large enough so that the set $\{x + \Delta S z \neq x \mid x \in \mathcal{X}, S \in \mathcal{S}, z \in \mathcal{Z}^{n_S}\}$ contains no points of $\mathcal{X}$. Therefore, if any Δ_k were as large as Δ then since all the trial points would lie outside while all the iterates lie inside $\mathcal{X}$, the k^{th} iteration must have been unsuccessful and the mesh size parameter reduced. Hence, Δ_k may never exceed $\Delta_0 \tau^{m_{max}}$, for some m_{max}. ∎

This lemma, combined with the assumption that all iterates lie in a compact set, is sufficient to show the following result. Its proof is omitted since it is identical to that of the same result in Torczon [15].

Theorem 3.2 *The mesh size parameters satisfy* $\liminf_{k \to +\infty} \Delta_k = 0$.

Since the mesh size parameter shrinks only at unsuccessful iterations, Theorem 3.2 guarantees that there are infinitely many unsuccessful iterations. Now we can specify the iteration subsequences we can show possess interesting properties:

Definition 3.3 *A convergent subsequence of iterates $\{x_k\}_{k\in K}$ (for some subset of indices K) of unsuccessful iterations is said to be a* refining subsequence *if* $\lim_{k\in K} \Delta_k = 0$.

Clearly there are refining subsequences, and we will show that the limit of any of them is an interesting point. We use $\hat{x}$ to denote a given limit point. It is our experience that using x_* or x^* to denote a limit point of a GPS iteration sometimes confuses people into thinking that the entire iteration converges, which is not generally true [1].

Lemma 3.4 *If $\hat{x}$ is the limit of a refining subsequence, and if s is any direction for which a poll step was evaluated for infinitely many iterates in the subsequence, and if f is Lipschitz in a neighborhood of $\hat{x}$, then the generalized directional derivative of f at $\hat{x}$ in the direction s is nonnegative, i.e., $f^\circ(\hat{x}; s) \geq 0$.*

Proof. Let $\{x_k\}_{k\in K}$ be a refining subsequence and $\hat{x}$ its limit point. ¿From Clarke [7], we have by definition that:

$$f^\circ(\hat{x}; s) = \limsup_{y\to\hat{x},\ t\downarrow 0} \frac{f(y+ts) - f(y)}{t} \geq \limsup_{k} \frac{f(x_k + \Delta_k s) - f(x_k)}{\Delta_k} .$$

First note that since f is Lipschitz near $\hat{x}$, it must be finite near $\hat{x}$. Note also that since a main point of the paper is to investigate the expedient of dealing with constraints by declining to evaluate f at infeasible points, we made the hypothesis that each term was actually evaluated infinitely many times. Thus, we have that infinitely many terms of the right hand quotient sequence is defined, and all of them must be nonnegative or else the corresponding poll step would have been successful (recall that refining subsequences are obtained from unsuccessful iterations). Of course, there may be no such s if S_k were defined in a way incompatible with the geometry of the constraints. ∎

The preceding easy result is the key to our analysis. Before we add the complication of dealing with constraints, we give the following quick corollary, which strengthens Torczon's unconstrained result. In this corollary, we will assume still that f is Lipschitz near $\hat{x}$, and in addition, we will assume that the generalized gradient of f at $\hat{x}$ is a singleton. This is equivalent to assuming that f is strictly differentiable at $\hat{x}$, i.e., that $\nabla f(x)$ exists and $\lim\limits_{y\to x, t\downarrow 0} \frac{f(y+tw)-f(y)}{t} = \nabla f(x)^T w$ for all $w \in \Re^n$ (see [7], Proposition 2.2.1 or Proposition 2.2.4).

Theorem 3.5 *Let $\Omega = \Re^n$ and $\hat{x}$ be the limit of a refining subsequence. If f is strictly differentiable at $\hat{x}$ then $\nabla f(\hat{x}) = 0$.*

Proof. Again from [7], if f is strictly differentiable at $\hat{x}$, then for any direction $w \neq 0$, $f^\circ(\hat{x}; w) = \nabla f(\hat{x})^T w$. Now let $\hat{S}$ be any positive spanning set that is used infinitely many times in the refining subsequence, there must be at least one since $\mathcal{S}$ is finite. Then by Lemma 3.4, for each $s_i \in \hat{S}$, $0 \leq \nabla f(\hat{x})^T s_i$. Thus, if we write w as a nonnegative linear combination of the elements of $\hat{S}$, then we see immediately that $\nabla f(\hat{x})^T w \geq 0$. But the same construction for $-w$ shows that $-\nabla f(\hat{x})^T w \geq 0$ and so $\nabla f(\hat{x}) = 0$. ∎

3.1 Linearly Constrained Convergence Results

In this section, we will consider only the case where Ω is defined through a finite set of linear constraints. In order to prove the relevant optimality results, we will have to assume that $\mathcal{S}$, even though finite, is rich enough to generate POLL sets that conform to the geometry of the boundary of Ω. Furthermore, to apply our proof technique, we must ensure that the spanning sets that reflect this geometry get used infinitely many times as we converge to a point on the boundary. This implies a prescience with respect to the boundary geometry for points near the boundary, and it is a tall order. Fortunately, Lewis and Torczon [14] have built the exactly machinery we need to generate the positive spanning matrices $S_k \in \mathcal{S}$.

We pause to remind the reader that for $x \in \Omega$, the tangent cone to Ω at x is $T_\Omega(x) = \mathrm{cl}\{\mu(w - x) : \mu \geq 0, w \in \Omega\}$. The normal cone to Ω at x is $N_\Omega(x)$ and can be written as the polar of the tangent cone: $N_\Omega(x) = \{v \in \Re^n : \forall w \in T_\Omega(x),\ v^T w \leq 0\}$. It is the positive span of all the outwardly pointing constraint normals at x.

It would add unnecessary length to this paper to rewrite the careful construction given by Lewis and Torczon for $\mathcal{S}$ and the choice rule for S_k from $\mathcal{S}$ at each iteration (their notation for S_k is Γ_k). The construction is presented there quite succinctly in Section 8 where they consider implementation issues. They also mention difficulties inherent to degenerate constraints. We will use the a simpler abstracted version here. We will summarize the properties we need in the following definition.

Definition 3.6 *A finite set $\mathcal{S}$ of positive spanning sets for $\Re^n$* conforms to Ω *for some $\epsilon > 0$, if there is a mapping $S : \Omega \to \mathcal{S}$ such that for every y is in the boundary of Ω, with $\|y - x\| < \epsilon$, some subset $S_y(x)$ of columns of $S(x)$ generate $T_\Omega(y)$.*

This definition allows positive bases with fewer directions than the construction in Section 8 of [14]. Consider for example the linear programming problem given in [13]

$$\begin{array}{rr} \displaystyle\min_{x=(a,b)^T} & -a - 2b \\ \text{s.t.} & 0 \leq a \leq 1 \\ & b \leq 0. \end{array}$$

The optimal solution is $\hat{x} = (1,0)^t$. Their construction of the spanning set S_k near the optimal solution contains a total of four directions (none for the null space, two for the tangent cone, and two for its negative). Definition 3.6 does not say how to construct the spanning set, but it does allows us to use fewer directions: two, $(-1,0)^T$ and $(0,-1)^T$, for the tangent cone and only one more, for example $(1,1)^T$, to complete S_k into a positive spanning set.

With this definition, we are ready for our convergence result. Note that if $x \in \Omega$ is not near the boundary, then $S(x)$ need only provide a positive spanning set for $\Re^n$, which is completely sensible.

Theorem 3.7 *Let $\hat{x}$ be the limit of a refining subsequence. If f is strictly differentiable at $\hat{x}$, and if $\mathcal{S}$ conforms to Ω for an $\epsilon > 0$, then $\nabla f(\hat{x})^T w \geq 0$ for $w \in T_\Omega(\hat{x})$, and $-\nabla f(\hat{x}) \in N_\Omega(\hat{x})$. Thus, $\hat{x}$ is a KKT point.*

Proof. If $\hat{x}$ is interior to Ω, then the result is just Theorem 3.5, and so we can proceed directly to the case where $\hat{x}$ is on the boundary of Ω. Since $\mathcal{S}$ conforms to Ω for an $\epsilon > 0$, we have from Lemma 3.4, that $\nabla f(\hat{x})^T s \geq 0$ for every column s of $S_{\hat{x}}(\hat{x})$. But since every $w \in T_\Omega(\hat{x})$ is a nonnegative linear combination of the columns of $S_{\hat{x}}(\hat{x})$, $\nabla f(\hat{x})^T w \geq 0$. To complete the proof, we multiply both sides by -1 and conclude that $-\nabla f(\hat{x})$ is in $N_\Omega(\hat{x})$. ∎

Remark 3.8 *If f were only assumed to be Lipschitz near $\hat{x}$, then we could still conclude from Lemma 3.4, that $f^\circ(\hat{x}; s) \geq 0$ for every column s of $S_{\hat{x}}(\hat{x})$*

4 Concluding Remarks

This paper puts together algorithmic contributions by Lewis and Torczon, some observations of ours about what is really needed to obtain convergence of those algorithms, and elements of nonsmooth analysis set forth by Clarke. Clarke's analysis is perfectly suited to expose the powerful behavior of certain subsequences of the GPS iterates under weakened assumptions that more closely correspond to the class of real problems for which GPS is a likely choice. Since we appreciate good technical writing, we take this opportunity to acknowledge the work of these authors.

We believe that this analysis helps confirm that GPS methods for general constraints will not be based on the appealingly simple "barrier" strategy of placing a high function value on infeasible trial points. We have in preparation a paper suggesting and analyzing a GPS algorithm for general constraints based not on a single objective, but on the filter approach of Fletcher and his coauthors [9], [10] and [11].

References

[1] Audet C. (1998), "Convergence results for patterns search algorithms are tight," *TR98-24* Department of Computational & Applied Mathematics, Rice University, Houston Texas.

[2] Audet C. and Dennis J.E.Jr (1998), "On the convergence of pattern search algorithms with mixed variables," *TR99-02*, Department of Computational and Applied Mathematics, Rice University, Houston Texas.

[3] Booker A.J., Dennis J.E.Jr, Frank P.D., Serafini D.B., Torczon V. and Trosset M.W.(1999), "A rigorous framework for optimization of expensive functions by surrogates," *Structural Optimization* Vol.17 No.1, 1-13.

[4] Booker A.J., Dennis J.E.Jr, Frank P.D., Moore D.W. and Serafini D.B.(1999), "Managing Surrogate Objectives to Optimize a Helicopter Rotor Design - Further Experiments," AIAA Paper 98-4717, St. Louis, September 1998.

[5] Choi T.D., Eslinger O.J., Kelley C.T., David J.W. and Etheridge M.(1998), "Optimization of automotive valve train components with implicit filtering," to appear in *Optimization and Engineering.*

[6] Choi T.D. and Kelley C.T.(1999), "Superlinear convergence and implicit filtering," to appear in *SIAM Journal on Optimization.*

[7] Clarke, Frank H.(1990) "Optimization and Nonsmooth Analysis," SIAM Classics in Applied Mathematics Vol.5, Philadelphia.

[8] Conn A.R., Gould N.I.M. and Toint Ph.L.(1991) "A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds," *SIAM Journal on Numerical Analysis* 28, 545–572.

[9] Fletcher R. and Leyffer S.(1997), "Nonlinear Programming without a penalty function," Dundee University, Dept. of Mathematics, Report NA/171.

[10] Fletcher R, Leyffer S. and Toint Ph.L.(1998), "On the global convergence of an SLP-Filter algorithm," Dundee University, Dept. of Mathematics, Report NA/183.

[11] Fletcher R, Gould N.I.M., Leyffer S. and Toint Ph.L.(1999), "On the global convergence of trust-region SQP-Filter algorithms for general nonlinear programming," Department of Mathematics, FUNDP, Namur (B), Report 99/03.

[12] Hough P.D., Kolda T.G. and Torczon V.(2000), "Asynchronous parallel pattern search for nonlinear optimization," Sandia National Laboratories, Report SAND2000-8213.

[13] Lewis R.M. and Torczon V.(1999), "Pattern search algorithms for bound constrained minimization," *SIAM Journal on Optimization*, Vol.9 No.4, 1082-1099.

[14] Lewis R.M. and Torczon V.(1998), "Pattern search methods for linearly constrained minimization," *ICASE NASA Langley Research Center* TR 98-3, to appear in *SIAM Journal on Optimization.*

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