ACCELERATIONS FOR GLOBAL OPTIMIZATION METHODS THAT USE SECOND DERIVATIVE INFORMATION

by

William Baritompa

Department of Mathematics, University of Canterbury, Christchurch, New Zealand

and

Adele Cutler

Department of Mathematics and Statistics Utah State University, Logan, Utah

No. 82 March, 1993

Abstract – Two new improvements for the algorithm of Breiman & Cutler are presented. Better envelopes can be built up using positive definite quadratic forms. Better utiližation of first and second derivative information is attained by combining both global aspects of curvature and local aspects nearthe global optimum. The basis of the results is the geometric viewpoint developed by the first author and can be applied to a number of covering type methods. Improvements in convergence rates are demonstrated empirically on standard test functions.

Keywords – Global optimisation, deterministic, mathematical programming

Accelerations For Global Optimization Methods That Use Second Derivative Information

William Baritompa[1] and Adele Cutler

Department of Mathematics
University of Canterbury
Christchurch, New Zealand

Department of Mathematics and Statistics
Utah State University
Logan, Utah

Two new improvements for the algorithm of Breiman & Cutler are presented. Better envelopes can be built up using positive definite quadratic forms. Better utilization of first and second derivative information is attained by combining both global aspects of curvature and local aspects near the global optimum. The basis of the results is the geometric viewpoint developed by the first author and can be applied to a number of covering type methods. Improvements in convergence rates are demonstrated empirically on standard test functions.

Keywords — Global optimization, deterministic, mathematical programming

Mathematics Subject Classification (1991) 90C30, 65K05

1. Preliminaries

Introduction

The algorithm of Breiman and the second author [3] is a global optimization method for multimodal, multivariate functions for which derivatives are available. When used for minimization, it requires a lower bound on the eigenvalues of the Hessian. Geometrically this bound provides global information about the degree of curvature of the downward bending parts of the function's graph. This bound is used together with the gradient to construct a lower envelope of the function's graph built up of paraboloids tangent at the points of function evaluation (see Fig. 1). Successive function evaluations raise this envelope until the value of the global minimum is known to the required degree of accuracy. In [2] a variation of this method is described which ignores the gradient and uses a bound reflecting the local curvature of the graph at the global minimum (see Fig. 2). In this paper these methods will be referred to as *simple parabolically based algorithms* or *SPBA*.

This paper presents two new improvements. Firstly *SPBA* is generalized to handle more sophisticated envelopes built up of graphs of positive definite quadratic forms. Secondly a new combination of acceleration techniques from [1] and [2] is applied to this generalization of *SPBA* and other related algorithms. Both of these modifications

---

1 Partially supported by an University of Canterbury Erskine grant

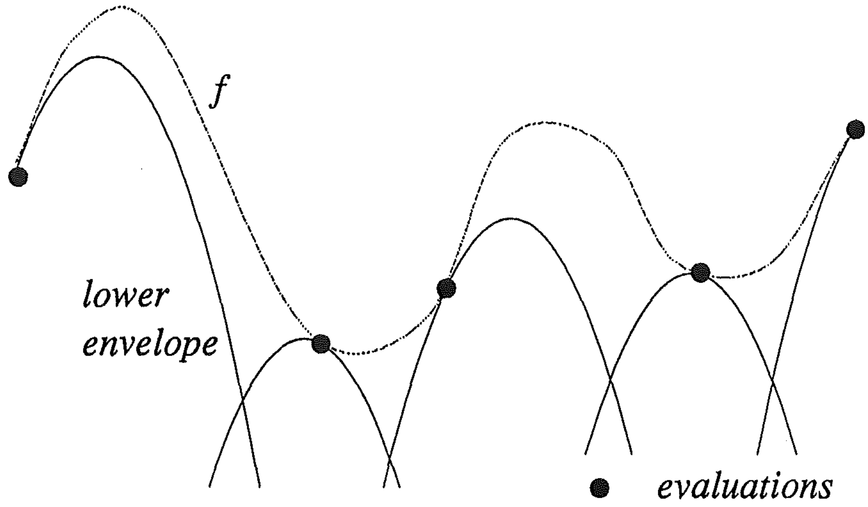

Figure 1 One dimensional illustration of *SPBA* using tangent parabolas to build lower envelope of function

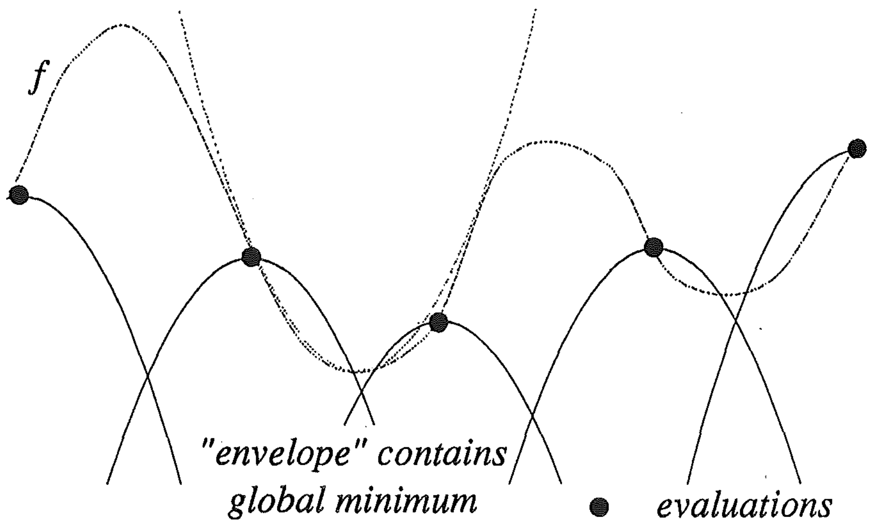

Figure 2 One dimensional illustration of *SPBA* ignoring the gradient using parabolas to build up an envelope containing the global minimum of f (since the lightly marked parabola fits over the global minimum, [2] showed, turned upside down, it can be used to build the envelope)

allow more detailed information about first and second derivatives to be utilized. In particular information about both the global nature of the downward bending parts of the graph and the local nature of the curvature at the global minimum are utilized effectively. The modifications are easily implemented, requiring only minor changes to the original implementation.

Zhigljavsky in [8] provides a good overview of currently known global optimization approaches. Table 1 of chapter 1 of [8] summarizes the prior information about the class of objective functions required for various approaches. The covering methods (of which *SPBA* is an example) are generally considered to require prior global smoothness conditions (such as (c') pg. 5 of [8] which concerns bounds on the Hessian). Smoothness conditions only in the vicinity of the global minimum ((o) pg. 7 of [8]) appear to be used primarily for random methods. It is worth noting this paper and [2] describe some covering methods appropriate to the latter prior conditions. Many of the ideas in this paper are applicable to covering methods in general.

This section continues with notation and background details. Section 2 provides the extensions and accelerations to the *SPBA*. Section 3 relates this to the acceleration results in [1]. Section 4 provides some comparison tests.

Notation and basic problem

This paper uses the same notation as [1]. The basic problem is to find the global minimum α and its location $E = f^{-1}(\alpha) \cap K$ of a function $f : K \to \mathbb{R}$ where $K \subset \mathbb{R}^n$ is a compact polytope. The *epigraph* of a function consists of all points on or above its graph.

Let ZG be the set of all differentiable functions with global minimum having zero gradient. Let $C_u^2(B_u)$ be the class of all twice differentiable functions such that $h(x_0 + \Delta x) = f(x_0) + \nabla f(x_0)\Delta x + \frac{1}{2}B_u\|\Delta x\|^2$ is an upper bound at each point of the domain x_0. Similarly let $C_l^2(B_l)$ have $h(x_0 + \Delta x) = f(x_0) + \nabla f(x_0)\Delta x - \frac{1}{2}B_l\|\Delta x\|^2$ as a lower bound at each point of the domain. For a given function the best bounds B_u and B_l, respectively, are the maximum and negative of the minimum of the eigenvalues of the Hessian. Let $L(M)$ be the class of Lipschitz continuous functions with constant M.

Background to the Simple Parabolically Based Algorithm

The following general framework due to Piyavskii [6] is useful for describing a number of algorithms including *SPBA*:

- *Initialization:*

  $\alpha_{-1} = \infty$

  $i = -1$

  Take a user specified x_0 from the domain K

- *Evaluation Step:*

  Increment i

  Compute function value, $f(x_i)$

  Compute gradient vector, $\nabla_i = \nabla f(x_i)$

  $\alpha_i = \min\{\alpha_{i-1}, f(x_i)\}$

- *Update Envelope Function Step:*

  Set $h_i(x) = h(x;\ x_i, f(x_i), \nabla_i)$ and let $F_i(x) = \max\limits_{k=0,\ldots,i} h_k(x)$

- *Get Next Sample Point Step:*

  $x_{i+1} = \arg\min\limits_{x \in K} F_i(x)$

- *Termination Test:*

  If $\min\limits_{x \in K} F_i(x)$ is close to α_i stop, otherwise go back to the evaluation step.

Provided $h_i(x) \leq f(x)$, the functions $F_i(x)$ are lower envelopes, and the global minimum is always between lowest value of the envelope, $\min_{x \in K} F_i(x)$, and the lowest known function evaluation, α_i. In this context, Piyavskii [6] showed that $\min_{x \in K} F_i(x)$ converges to α. Different choices of $h_i(x)$ determine specific algorithms [5, 7].

Within this framework two variations of simple parabolically based algorithms can be defined. Firstly let *SPBA with bound* B_l use $h_i(x) = f(x_i) + \nabla_i^T(x - x_i) - \frac{1}{2}B_l\|x - x_i\|^2$. The constant B_l must be chosen so $h_i(x) \leq f(x)$. This gives the algorithm of Breiman and the second author [3] illustrated by Fig. 1. Secondly (see Fig. 2) let *SPBA with bound* B_u *and zero linear term* use $h_i(x) = f(x_i) - \frac{1}{2}B_u\|x - x_i\|^2$. Provided $f(x\mathrm{gm}) + \frac{1}{2}B_u\|x - x\mathrm{gm}\|^2 \geq f(x)$ for all x in the domain (here x_{gm} is the location of the global minimum), proposition 3.2 in [2] shows the method will work. Note, as remarked in [2], this does not lead to a lower envelope for f, however, the global minimum of $F_i(x)$ still provides a lower bound for the global minimum of f. Remark 5.3 of [2] observes the implementation of [3] works in this case by the simple expediency of taking the gradient always to be the zero vector.

Geometrically, the set of points above or on the graph of $F_i(x)$ and below or on the hyperplane at height α_i form a bracket of the point(s) on the graph of f corresponding to the global minimum. In [3] the bracket is not explicitly used, however, updating the envelope and finding the arg min can be viewed as dealing with the bracket. The bulk of the work in the implementation of *SPBA* is at the *Get Next Sample Point* step, because this step is potentially as difficult as the original problem. Specific mathematical properties of $h_i(x)$ facilitate efficient implementation. The idea in [3] is to keep track of all the local minima of the lower envelope, so the next sample point is the lowest of these local minima. Around the i^{th} sample point is a region over which the envelope is $h_i(x)$. Since $h_i(x) - h_j(x)$ is linear, this region is a polytope. Since $h_i(x)$ is concave, the local minima of the envelope are located at vertices of the collection of polytopes. The implementation in [3] keeps track of the vertices and edges of the polytopes. Updating the vertex structure entails removing those vertices which are no longer needed and finding the vertices of the new polytope. Since $f(x_{i+1}) \geq F_i(x_{i+1})$, the vertices to be removed can be found by moving along the edges of the polytopes.

Intuition behind the two versions of *SPBA*

The two versions of *SPBA* take advantage of completely different information. Intuitively both versions of *SPBA* build up the bracket from parabolic pieces. Blunt pieces work faster than sharp ones which mean the smaller the constants used by *SPBA* the better. For functions with an interior global minimum, the first version of *SPBA* works well when the downward bending parts of the graph are gently curved, while the second version (with zero linear term) works well on functions that are gently bending upwards at the global minimum.

Background to Accelerations

The geometric viewpoint developed in [2] is the key behind the acceleration ideas presented in this paper. The viewpoint is that the bracket found by the algorithm occurs by removal of certain regions at each step. Modifications of an algorithm to use bigger removal regions produce accelerations. This is the basis of propositions 2 and $2'$ in this paper.

The approach for describing the accelerations developed in [1] is used in this paper. It concerns the way the next sample point is used by the algorithm during the *Update Envelope Function* step. Replacement values x_i^a and $f^a(x_i)$ which are easily computed from $x_i, f(x_i)$ and ∇_i are used to compute $h_i(x)$. Faster convergence results and the minimal extra computation does not affect the overheads of the algorithm.

2. Modifying the Simple Parabolically Based Algorithm

Generalization to Use Arbitrary Paraboloids

Referring to the general algorithm description in section 1, observe for *SPBA* that $h_i(x)$ $= f(x_i) + L(x - x_i) - \frac{1}{2}q(x - x_i)$ where $L = \nabla_i^T$ or $L = 0$ and $q(x) = B\|x\|^2$. Other quadratic forms can be used to produce convergent methods:

- Let $L = \nabla_i^T$ and use any quadratic form, $q(x) = x^T H x$, such that $h_i(x) \leq f(x)$ holds. Piyavskii's condition guarantees convergence.
- Use $L = 0$ and a quadratic form, $q(x) = x^T G x$, satisfying $f(x\text{gm}) + \frac{1}{2}q(x - x\text{gm}) \geq f(x)$ for all x in the domain (here x_{gm} is the location of the global minimum). As before, proposition 3.2 in [2] shows the method will work although it does not lead to a lower envelope for f.

The *SPBA* can be seen geometrically as removing regions which are translates of the epigraph of the quadratic form $-\frac{1}{2}x^T D x$ where $D = BI$ is a diagonal matrix with the second derivative bound B on the diagonal. These regions are paraboloids with spherical horizontal cross sections. As noted in the bulleted remarks above, other quadratic forms give valid methods. We introduce two versions of a *General Parabolically Based Algorithm.* Let *GPBA with P* be the same as *SPBA* except that $h_i(x) = f(x_i) + \nabla_i^T(x - x_i) - \frac{1}{2}q(x - x_i)$. Let *GPBA with P and zero linear term* be the same as *SPBA* except that $h_i(x) = f(x_i) - \frac{1}{2}q(x - x_i)$. In both cases $q(x) = x^T P x$. Interestingly, the implementation of *SPBA* described in [3] works if P is any positive definite matrix, so the algorithm handles removal of arbitrary paraboloids.

Proposition 1 *Let H and G be positive definite matrices as described above. The implementation of SPBA, with only the formula for $h_i(x)$ changed as above, realizes (1) GPBA with H and (2) GPBA with G and zero linear term.*

> **Proof:** The two requirements of Theorem 3.1 in [3] are $h_i(x)-h_j(x)$ is linear in x and $h_i(x)$ is concave. It is easy to verify that both of these conditions hold for (1) and (2). Additionally for efficient updating of the data structure ("finding the dead vertices") it is required that $F_i(x_{i+1}) \leq f(x_{i+1})$. Since $x_{i+1} = \arg\min F_i(x)$ and $F_i(x)$ determines a bracket for the global minimum, we have $F_i(x_{i+1}) \leq \alpha$, and the required inequality holds. ■

While the versions of *SPBA* require bounds on the eigenvalues of the Hessian, proposition 1 shows how more detailed information about the Hessian can be used. The intuition discussed earlier is extended here, H reflects the curvature of the downward bending parts of the graph, while G reflects the upward bending part of the graph at the global minimum. Empirical tests in section 4 show good choices of H and G make *GPBA* perform better than *SPBA* using the best possible bounds.

Accelerations Found By Combining Regions

We show that the two types of regions relating to H and G can be combined to form a better region and thus take advantage of both aspects relating to the graphs curvature. Fortuitously this new region is also the translate of the epigraph of a positive definite quadratic form and can be handled by *GPBA*.

The following key lemma provides the details. It describes the effect of sliding the epigraph of one positive definite quadratic form along the graph of another. Fig. 3 illustrates this in the one dimensional case.

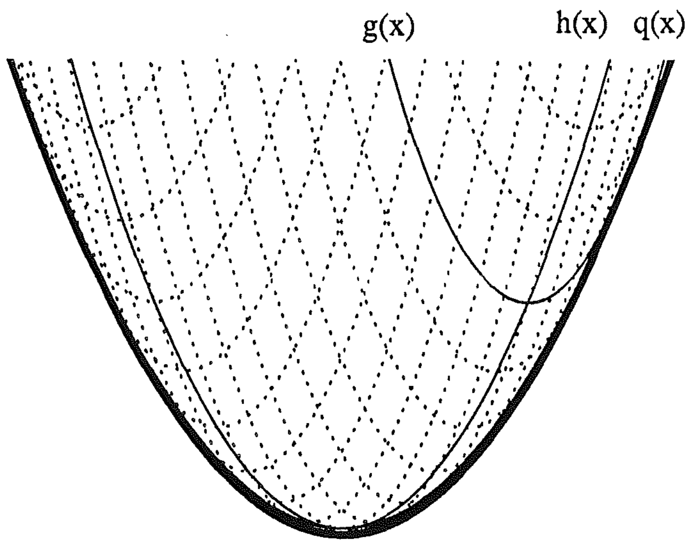

Figure 3 Sliding one paraboloid along another is enveloped by a paraboloid

Lemma 2 *Let H and G be positive definite matrices. Let $S = (H+G)^{-1}$ and $Q = H^T S^T G S H + G^T S^T H S G$. The epigraph of the quadratic form $x^T Q x$ is obtained by sliding the epigraph of the quadratic form $x^T G x$ along the graph of the quadratic form $x^T H x$.*

Proof: Let g and h be the two quadratic forms represented by G and H respectively. Let $q(x)$ be the lower envelope obtained by sliding the graph of g along h. We now show $q(x)$ is a quadratic form with the required matrix.

$$\begin{aligned} q(x) &= \min_y \left(g(x-y) + h(y)\right) \\ &= \min_y \left((x-y)^T G(x-y) + y^T H y\right) \end{aligned}$$

Taking derivatives with respect to y, the minimum occurs when $2G(x-y) = 2Hy$. Since H and G are positive definite, the sum is invertible, thus $y = (H+G)^{-1}Gx$. Using this value for y, gives $q(x) = x^T Q x$ with $Q = H^T S^T G S H + G^T S^T H S G$ as required. ■

The following proposition shows that *GPBA* with Q from Lemma 2 and zero linear term can be used to give accelerated performance.

Proposition 2 *Given a function f and positive definite matrices H and G as described. Let $S = (H+G)^{-1}$ and $Q = H^T S^T G S H + G^T S^T H S G$. At each iteration calculate replacement values $x_i^a = x_i + H^{-1}\nabla f(x_i)$ and $f^a(x_i) = f(x_i) + \frac{1}{2}(\nabla f(x_i))^T H^{-1} \nabla f(x_i)$. An acceleration over both (1) and (2) of proposition 1 is obtained by using the replacement values during the Update Envelope Function step of GPBA with Q and zero linear term.*

Before providing the proof, it is worth looking at a few special cases. If H and G commute, the formula simplifies to $Q = HG(H+G)^{-1}$. For $H = B_l I$ and $G = B_u I$, the formula gives $Q = \dfrac{B_l B_u}{B_l + B_u} I$. We state this special case and note that the replacement values are appropriate to other algorithms like [4] which use the same building blocks for the lower envelope.

Proposition 2′ *Given a function f in the class $C_l^2(B_l) \cap C_u^2(B_u) \cap ZG$. Let $B = B_l B_u/(B_l + B_u)$. An acceleration is obtained by using the replacement values x_i^a and $f^a(x_i)$ during the Update Envelope Function step of SPBA with B and zero linear term. Here $x_i^a = x_i + (1/B_l)\nabla f(x_i)$ and $f^a(x_i) = f(x_i) + \dfrac{\|\nabla f(x_i)\|^2}{2B_l}$.*

The proof using the geometric ideas developed in [1] and [2] identifies a bigger region that can be removed.

Proof of proposition 2: (Refer to Fig. 4). Given H, G and Q as stated. Let h, g and q be the corresponding quadratic forms respectively. Let $x_i^a =$

$x_i + H^{-1}\nabla f(x_i)$ and $f^a(x_i) = f(x_i) + \frac{1}{2}(\nabla f(x_i))^T H^{-1}\nabla f(x_i)$ as stated. During the *Update Envelope Function* step of *GPBA* with H, the function $h_i(x)$ is $f(x_i) + \nabla_i^T(x - x_i) - \frac{1}{2}h(x - x_i)$. Expressed in terms of the replacement values this is $f^a(x_i) - \frac{1}{2}h(x - x_i^a)$. Now H was chosen so $h_i(x) \leq f(x)$. So by proposition 3.2 of [2] at all points of the graph of $h_i(x)$, the regions below the translated graphs of $-\frac{1}{2}g(x)$ can be removed, the union of these by Lemma 2 is the region below $f^a(x_i) - \frac{1}{2}q(x - x_i^a)$. It is a paraboloid with maximum at $(x_i^a, f^a(x_i))$ and contains the paraboloids used by the two methods mentioned in Proposition 1. So using the replacement values for *GPBA* with Q and zero linear terms gives an acceleration over either method (1) or (2) of proposition 1. ■

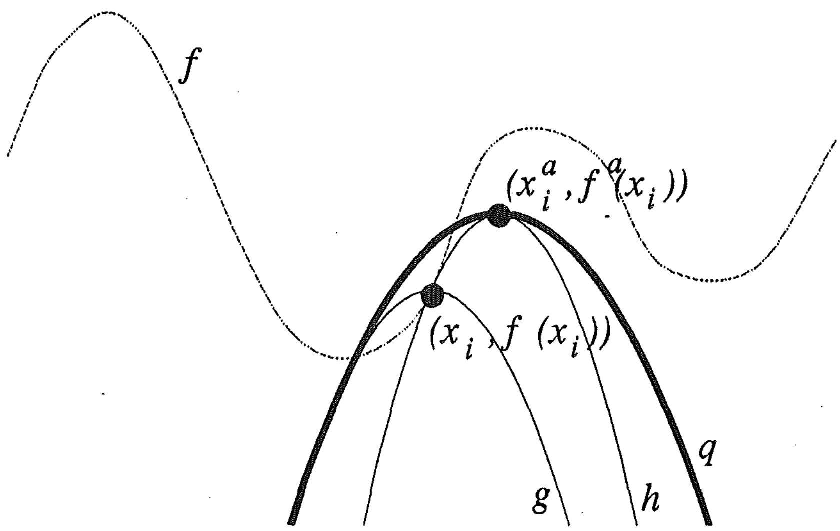

Figure 4 One dimensional illustration for proof of Proposition 2

3. Relation to other algorithms and accelerations

If a Lipschitz bound M is available, the accelerations given in [1] are compatible with proposition $2'$.

A Lipschitz bound can be used in conjunction with second derivative bounds by *SPBA* and related methods such as [4]. Note in the case of the Lipschitz bound going to infinity, the following reduces to proposition $2'$.

Proposition 3 *Given a function f in the class* $C_l^2(B_l) \cap C_u^2(B_u) \cap L(M) \cap ZG$. *Let* $B = B_l B_u/(B_l + B_u)$. *An acceleration is obtained by using replacement values* x_i^a *and* $f^a(x_i)$ *during the Update Envelope Function step of SPBA with B and zero linear term. Here* $x_i^a = x_i + (1/B_l)\nabla f(x_i)$ *and*

$$f^a(x_i) = \begin{cases} f(x_i) + \dfrac{\|\nabla f(x_i)\|^2}{2B_l} + \dfrac{B}{2M^2}(d_i - l_i)^2 & d_i > l_i \\ f(x_i) + \dfrac{\|\nabla f(x_i)\|^2}{2B_l} & d_i \le l_i \end{cases} \quad \text{where}$$

$$d_i = f(x_i) - \alpha_i \text{ and } l_i = \frac{M^2}{2B} - \frac{\|\nabla f(x_i)\|^2}{2B_l}.$$

Proof: The paraboloid which is the region below the graph of $f^a(x_i) - \frac{1}{2}B\|x - x_i^a\|^2$ is removed during the i^{th} step of *SPBA*. Using x_i^a and $f^a(x_i)$ in place of x_i and $f(x_i)$ in proposition 5 of [1] gives the result. ■

Algorithms using a Lipschitz bound can modified to incorporate second derivative bounds and use the gradient. The algorithm of Mladineo [5] deals with Lipschitz continuous function in $L(M)$. It is an algorithm in Piyavskii's scheme with $h_i(x) = f(x_i) - M\|x - x_i\|$ and for dimension one reduces to that of Piyavskii [6] and Shubert [7].

Proposition 4 *Given a function f in the class* $C_l^2(B_l) \cap C_u^2(B_u) \cap L(M) \cap ZG$. *Let* $B = B_l B_u/(B_l + B_u)$. *When using the algorithm of Mladineo, an acceleration using gradient information is obtained by using the replacement sample point* $(x_i^a, f^a(x_i))$ *during the Update Envelope Function step. Its coordinates are given by*

$$x_i^a = x_i + \frac{1}{B_l}\nabla f(x_i) \text{ and } f^a(x_i) = \begin{cases} \alpha_i + \dfrac{M}{\sqrt{B}}\sqrt{2d_i} & d_i < \frac{M^2}{2B} \\ f(x_i) + \dfrac{M^2}{2B} + \dfrac{\|\nabla f(x_i)\|^2}{2B_l} & d_i \ge \frac{M^2}{2B} \end{cases}$$

where $d_i = f(x_i) - \alpha_i + \dfrac{\|\nabla f(x_i)\|^2}{2B_l}$.

Proof: In the notation of [1] and using a slight extension of Lemma 2, it follows that the *MB*-parabolically capped cone with apex at $(x_i^a, f^a(x_i))$ could be removed during the i^{th} step of Mladineo's algorithm. Using x_i^a and $f^a(x_i)$ in place of x_i and $f(x_i)$ in proposition 3 of [1] gives the result. ■

4. Comparisons

The results of this paper provide accelerations in a one-step sense. For a given iteration, using an acceleration always produces a better bracket than not using it. Since the sequence of sample points used by these algorithms is determined by using the arg min $F_i(x)$ at each step, using the accelerations will produce a different sequence of sample points, so on occasion will perform worse. We explore the convergence behaviour empirically.

Testing

RATFOR implementations of *SPBA* and *GPBA* were run with the various modifications for a number of standard test functions. Each iteration requires a function and gradient evaluation. Tests were stopped when both an absolute error measure was less than 0.01 and a relative measure was less than 0.0001. The tests of the acceleration of Mladineo's algorithm were carried out only by a discrete simulation (described in [2]). The particulars are summarized in Tables 1–4.

The standard test functions are described in [3]. To illustrate the differences between using G and H, variants a-d of EXP2 of the form $f(x,y) = -\pi e^{-1/2\left(ax^2+b(y-e)^2\right)} - (1-\pi)e^{-1/2\left(cx^2+d(y+e)^2\right)}$ were used. EXP2 has circular contours, the variants have eliptical contours at the global minimum. EXP2b is highly curved at the global minimum. EXP2c and d have two local minimum. The contours around both these look similar for EXP2c while they are quite different for EXP2d. Table 5 gives details.

Test parameters	$L = \nabla_i^T$		$L = 0$		$L = 0$	
	B_l	M	B_u	M	$B_lB_u/(B_l+B_u)$	M
Reference	[3]	[1] p6	[2] rmk5.3	[1] p5	prop 2′	prop3
EXP2	27	29	10	9	7	7
COS2	77	72	62	60	34	34
RCOS	265	263	306	309	142	142
GW	939	632	905	601	445	336
C6	111	111	>1000	>1000	105	107

Table 1 ITERATIONS TO CONVERGENCE USING SPBA AND ITS ACCELERATIONS
Heavier shaded columns from methods presented in earlier references.

Test parameters	$L = \nabla_i^T$		$L = 0$		$L = 0$	
	B_l	H	B_u	G	$B_lB_u/(B_l+B_u)$	Q
Reference	[3]	prop1	[2] rmk5.3	prop1	prop 2′	prop2
EXP2	27	27	10	10	7	7
EXP2a	44	30	53	25	25	11
EXP2b	147	82	261	127	88	44
EXP2c	76	47	57	30	33	17
EXP2d	50	45	49	41	23	19

Table 2 ITERATIONS TO CONVERGENCE USING SPBA AND GPBA (shaded) AND ITS ACCELERATIONS

Test	Mladineo [1] prop 4	Mladineo prop 4	SPBA prop 3
EXP2	19	8	7
COS2	61	39	34
RCOS	176	123	100
GW	434	282	283
C6	52	55	79

Table 3 ITERATIONS TO CONVERGENCE (DISCRETE TESTS)
Heavier shaded column from method presented in earlier reference.

Test	Domain	Initial Point	Lipshitz constant M	B_u	B_l	B
EXP2	$(-1,1) \times (-1,1)$	$(0.2, 0.2)$	0.61	1	0.37	0.269
COS2	$(-1,1) \times (-1,1)$	$(0.5, 0.5)$	4.8	26.7	22.7	12.26
RCOS	$(-5,10) \times (0,15)$	$(0,5)$	113.6	29.2	16.8	10.65
GW	$(-100,100) \times (-100,100)$	$(25,25)$	2.15	1.01	0.99	0.500
C6	$(-5,5) \times (-5,5)$	$(0,0)$	5601	5628	8.93	8.92

Table 4 Particulars for test functions (best bounds used)

Test	π	a	b	c	d	e	B_u	B_l	G	H
EXP2	1	1	1	-	-	0	1	0.37	1.0 I	.37 I
EXP2a	1	4	16	-	-	0	16	7.14	1.0 D	.446 D
EXP2b	1	40	160	-	-	0	160	71.2	10 D	4.45 D
EXP2c	.55	4	16	4	16	.5	8.77	6.51	.58 D	.407 D
EXP2d	.55	4	16	16	4	.5	8.07	4.23	.82 D	.399 E

Table 5 VARIANTS OF THE EXP FUNCTIONS (all using same initial point and domain as EXP2)
(note: D = diag(4,16), E = diag(9.4, 10.6))

Comments

Five conclusions are apparent.

- Proposition 2, $2'$ and 3 always provide substantial improvements. The number of iterations in right hand columns of Tables 1 and 2 are often half the size of the corresponding entries in the first two columns.
- *GPBA* using other positive definite matrices often shows marked improvement over *SPBA* using multiples of the identity as seen by comparing shaded and unshaded entries in Table 2.

- EXP2a and EXP2b reflect the differences in using G and H (rows 2 and 3 of Table 2). EXP2a is gently curved at the global minimum so using G is better than H, while EXP2b is strongly curved and H works better.
- The use of the Lipschitz constant M (the darker halves of the columns in Table 1) usually has no effect. Those accelerations using M take effect only if there is a large drop in value, and thus help only in the early stages of an algorithm. For RCOS and C6 this minimal drop is nearly the overall distance from minimum to maximum, so acceleration hardly occurred. For GW the minimal drop is quite small, the replacement values were often used, and the improvement is quite marked. Note sometimes these "accelerations" produced marginally poorer results. This is due to the fact that different sample sequences were produced. When repeated trials averaged over many different initial points were done, the accelerations were never worse.
- Both Mladineo's and the second author's algorithms when fully utilizing first and second derivative information give very similar results as shown by the similarity of columns 2 and 3 in Table 3

Concerning the discrete tests done for Mladineo's algorithm, note column 3 of Table 3 and the shaded part of column 3 of Table 1 test the same method. Likewise the heavily shaded results of Table 1 were done with discrete testing in [1]. The values are comparable and confirm that the discrete testing gives similar results to the actual running of the algorithms. The problem relates to differences in the stopping criterion. Discrete testing is appropriate for comparison testing shown in Table 3.

Conclusions and Future Directions

We have demonstrated two ways of improving the performance of some global optimization methods. The algorithms were easily modified to utilize fully both first and second derivative information.

A drawback of many methods that use bounds on first or second derivatives, including the ones presented here, concerns the calculation of the bounds. Finding good ones is often an equally difficult global optimization as the original. Work in this direction is needed. Local bounds appropriate to small regions in the domain are sometimes easier to obtain. So one area for future work appropriate to *GBPA* concerns incorporating subdivision of the domain, a modification that would readily lend itself to parallel computing.

References

[1] W. Baritompa, Accelerating Methods for Global Optimization, submitted to *J. Global Optimization*

[2] W. Baritompa, Customizing methods for global optimization — a geometric viewpoint, *J. Global Optimization* To appear

[3] Breiman & Cutler, A Deterministic Algorithm for Global Optimization, *Math. Program.* To appear

[4] Yu. G. Evtushenko (1971), Numerical methods for finding global extrema (case of a non-uniform mesh), *USSR Comp. Math. and Math. Phys.* **11**, 6, 1390–1403

[5] R. H. Mladineo (1986), An algorithm for finding the global maximum of a multimodal, multivariate function, *Math. Program.* **34** 188–200.

[6] S.A. Piyavskii (1972), An algorithm for finding the absolute extremum of a function, *USSR Comp. Math. and Math. Phys.* **12**, 57–67

[7] Bruno O. Shubert (1972), A sequential method seeking the global maximum of a function, *SIAM J. Numer. Anal.* **9**, 379–388

[8] Anatoly A. Zhigljavsky, editor J. Pinter (1991), Theory of Global Random Search, *Series: Mathematics and its applications (Kluwer Academic Publishers)*