EQUIVALENT METHODS FOR GLOBAL OPTIMIZATION

by

DIANE MACLAGAN and TIMOTHY STURGE
WILLIAM BARITOMPA
Department of Mathematics and Statistics
University of Canterbury, Christchurch, New Zealand

No. 122 March, 1995

Abstract – The envelope used by the algorithm of Breiman and Cutler [4] can be smoothed to create a better algorithm. This is equivalent to an accelerated algorithm developed by the third author and Cutler in [3] using envelopes which seemed poor ones at first sight. Explaining this anomaly lead to a general result concerning the equivalence of methods which use information from more than one point at each stage and those that only use the most recent evaluated point. Smoothing is appropriate for many algorithms, and we show it is an optimal strategy.

Equivalent Methods for Global Optimization *

DIANE MACLAGAN AND TIMOTHY STURGE

WILLIAM BARITOMPA wpb@math.canterbury.ac.nz

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

Received March 2, 1995

Editor:

Abstract. The envelope used by the algorithm of Breiman and Cutler [4] can be smoothed to create a better algorithm. This is equivalent to an accelerated algorithm developed by the third author and Cutler in [3] using envelopes which seemed poor ones at first sight. Explaining this anomaly lead to a general result concerning the equivalence of methods which use information from more than one point at each stage and those that only use the most recent evaluated point. Smoothing is appropriate for many algorithms, and we show it is an optimal strategy.

Keywords: Global Optimization, deterministic, algorithms, optimality

1. Introduction

Many global optimization algorithms use an auxiliary function, upper envelope, or some variation of this concept to determine the next evaluation point. One way to improve such an algorithm is to improve the upper envelope. However, we were aware of some algorithms by the third author that used very strange auxilary functions but behaved very well. In trying to understand this seeming mystery, we found they were mathematically equivalent to some other (but slightly more difficult to implement) algorithms using nice envelopes.

Smooth envelopes approximate smooth functions well, and were used by Sergeyev to get an improved algorithm [8]. Section 2.4 describes, in a slightly more general context, a version of Breiman and Cutler's algorithm [4] using smooth envelopes. This smoothed variation has an optimal envelope. Section 5 shows using envelopes in the class of functions under consideration always gives an optimal envelope.

In section 3 we show the smoothed method above is equivalent to a seemingly quite different accelerated version described in [3]. This is a special case of a more general result (in section 4) about the equivalence of algorithms which use information from more than one point to improve an existing envelope at each stage, and those that only use the most recent evaluated point.

* Part of an Honours III project by the first two authors under the supervision of the third.

2. Preliminaries

We consider the following one dimensional unconstrained global optimization problem: given a function $f : [a, b] \subseteq \mathbb{R} \to \mathbb{R}$ determine the points where the function f attains its maximal value (or within ϵ of it). We require a algorithmic procedure based on the sequential evaluation of points in the domain. The algorithm is to be applied to a restricted class of functions (e.g. Lipschitz continuous functions, Lipschitz continuous derivatives etc). Although we consider only the case for functions of one variable, many of the ideas are applicable for functions of several variables.

2.1. Definitions

Let X^* be the set of points in the domain where the maximum value is attained and let α denote the maximum value.

An *upper envelope* of a function $f : [a, b] \to \mathbb{R}$, is a function $g : [a, b] \to \mathbb{R}$ such that $\forall x \in [a, b],\ \ f(x) \leq g(x)$. Given an upper envelope g, a local maximum on its graph is called a *peak point* of g. The set of all these points is the *peak set*. A subset of the domain related to this is the *highest set*, $\{x | g(x) = \max_{[a,b]} g\}$.

A function g is called a *pseudo upper envelope* for a function f if $g(x) \geq f(x) \, \forall \, x \in X^*$. This means that all the global maxima of f lie below the graph of g.

Figure 1 explains the terms in a graphical manner. Here f has been evaluated at points labeled 1 through 4, and the peak points of g are circled. In practice an upper bound for the *error* in the estimate of the global value is used in the stopping criteria. *Variation* as shown in the figure is a common one.

2.2. A general deterministic algorithm

The following framework encompasses many standard algorithms.

1. `Initialization: Input global parameters. Choose initial points and make evaluations.`

2. `Build new envelope: Using the global parameters and all previous information form a new (pseudo) envelope` $g(x)$.

3. `Next point strategy: Choose a new point from the highest set of` g.

4. `Evaluation: Evaluate the function and any needed derivatives at the new point.`

5. `Stop if suitable criteria are met or loop.`

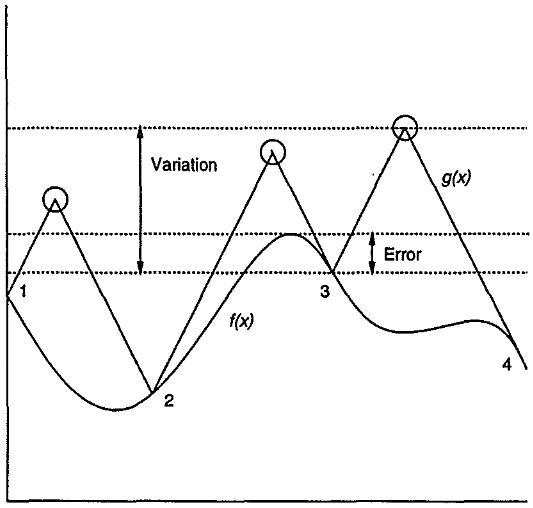

Figure 1. A graphical example of the definitions

2.3. One point and multi-point methods

There are two distinct methods of building the new envelope in the general algorithm. Sometimes we generate the new upper envelope by improving the old one based *only* on information from the new evaluated point. We call this a one point method. The second method is to use information from more than one evaluated point to build up the new upper envelopes at each stage, which we call a multi-point method. This paper presents some results concerning the equivalence of one and two point methods.

A common one point method produces the new envelope in the following way: $g_{n+1}(x) = \min(\phi_n(x), g_n(x))$, where $\phi_n(x)$ depends only on the new information. When the functions ϕ_n are all translates one functions ϕ, we call ϕ the *cutter*, and the algorithm a *cutting algorithm* [1]. A cutting algorithm sequentially removes regions where the global optima is known not to be. This is used in the algorithms such as those developed by Piyavskii [7], Shubert [9], and Breiman and Cutler [4]. For the first and second of these, the cutter is $\phi(x) = m|x|$ for Lipschitz constant m, and for the third it is $\phi(x) = \frac{1}{2}sx^2$ for second derivative bound s.

One point methods have the advantage that it is relatively easy to extend them to more than one dimension, although the problem of determining the maximal value of the envelope becomes much more difficult.

An example of an algorithm using information from two points is the modified version of Brent's [5] algorithm described in [1]. In the case of two point methods, the section of the envelope between two adjacent points is replaced by a better one.

Note in this case the upper envelope is not the minimum of several other upper envelopes, but instead of curve sections between evaluated points. The envelope thus produced is usually a function in the correct class on the interval between the two points.

These algorithms appear to be more difficult to generalise to higher dimensions, as the concept of 'adjacency' is lost. Sergeyev [8] and Pinter [6] use an diagonal partitioning scheme to utilize one dimensional auxiliary functions in higher dimensions. Multi-point methods are appropriate here but this is not a direct generalisation.

These two methods are contrasted in figure 2. In the diagram on the left, one curve is placed at each evaluated point, while in the diagram on the right, a curve is placed through each adjacent pair of evaluated points.

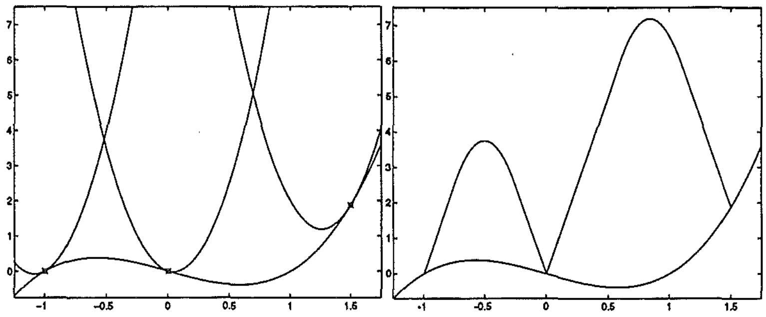

Figure 2. An example of a one point (left) and a two point (right) method

2.4. A smoothed variation of the algorithm of Breiman and Cutler

The algorithm of Breiman and Cutler [4] (denoted here by BC) is a multidimensional algorithm. We give a brief one-dimensional overview. BC assumes $f \in C^2$ and a one sided bound on the second derivative, $f''(x) \leq s_u \quad \forall x \in [a, b]$. Each iteration uses the function value and gradient to fit a parabola of second derivative s_u tangent to the evaluated point.

As can be seen from diagram 3 in BC the upper envelope produced does not have a bounded second derivative. If there is a two sided bound $-s_l \leq f''(x) \leq s_u \quad \forall x \in [a, b]$ (or $|f''(x)| \leq s$), then this method is open to the aesthetic criticism that the upper envelope is not like the function being optimized, so cannot be a good approximation to the function at the global optimum. So for two sided bounds an improved algorithm should be possible. This has been done in two ways.

In [3] this additional information provides an acceleration to BC by placing a parabola with second derivative $(s_l s_u)/(2(s_l + s_u))$ at the vertex of the original parabola (see figure 4). This improvement (referred to here as accelerated BC)

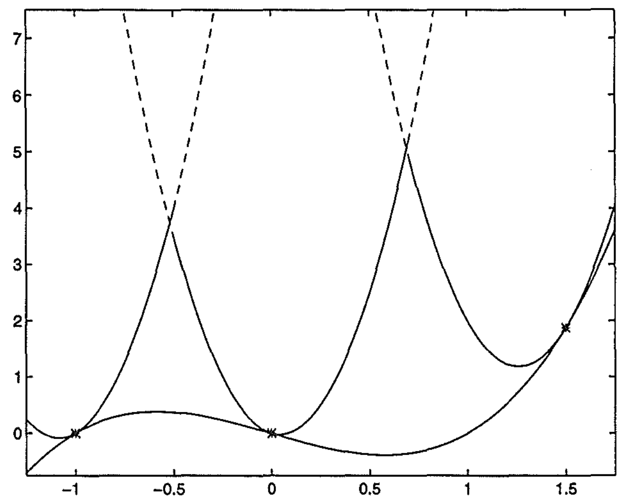

Figure 3. The upper envelope of Breiman and Cutler's method

forms a pseudo upper envelope rather than an upper envelope. It is still subject to the same aesthetic criticisms of the original BC, as well as the the added criticism of being only a pseudo upper envelope.

We approach the two sided bound on the second derivative from a different angle. We consider the more general Lipschitz condition on the first derivative: $-s_l(x-y) \leq f'(x)-f'(y) \leq s_u(x-y) \forall x > y,\ x, y \in [a,b]$. We develop a 'smoothed' variation of BC (referred to as smoothed BC) by placing a parabola tangent to the original two parabolas, while still retaining an upper envelope (see figure 5). A similar method has been developed independently by Sergeyev in [8]. The justification being that the smoothness of the envelope means that it is closer to the function being optimized. In fact the envelope formed is optimal, as is discussed in section 5, because it is in the class of functions being optimized.

3. The equivalence of smoothed and accelerated BC

We had originally believed that smoothed BC would further improve accelerated BC. This did not happen, however, as the peak points of the accelerated BC pseudo envelopes lie were the local maxima of the smoothed BC upper envelopes (see figure 6). This means the peak sets created in the two algorithms are identical so the sequence of evaluated points is identical. Thus smoothed BC is effectively *the same as* accelerated BC in one dimension. The following theorem proves this claim.

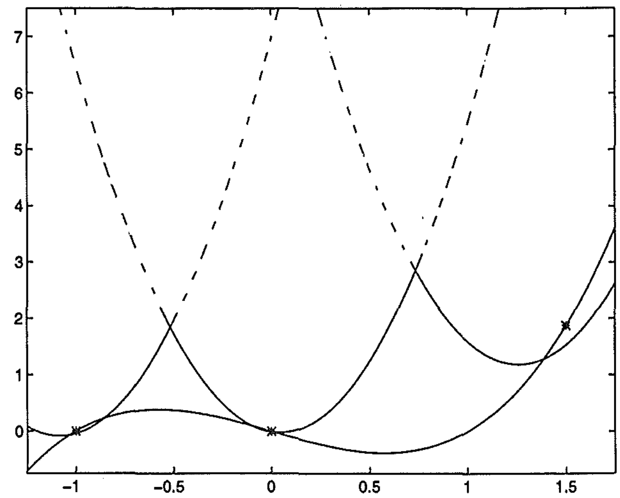

Figure 4. The pseudo upper envelope for accelerated Breiman and Cutler

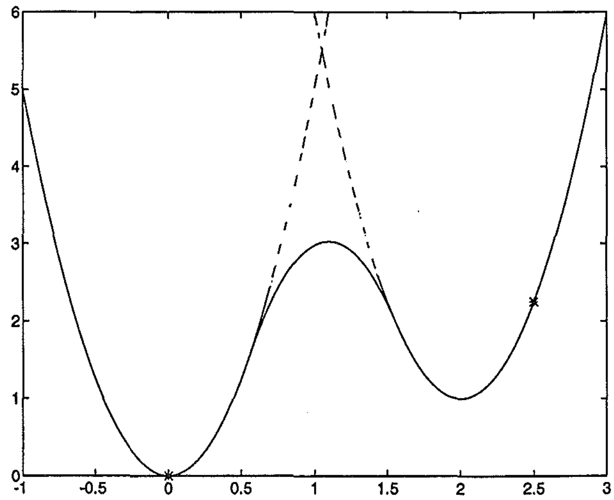

Figure 5. The upper envelope for smoothed Breiman and Cutler

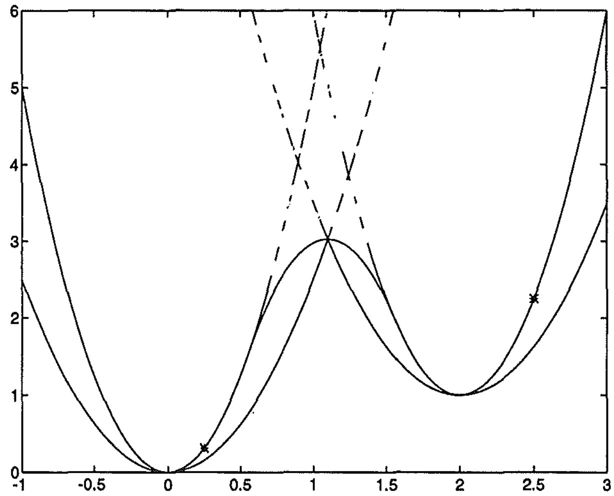

Figure 6. Envelopes of accelerated and smoothed BC have the same peak points

Theorem 1 *The smoothed BC algorithm creates the same peak set as the accelerated BC algorithm in one dimension.*

Proof: We show that the maxima of the envelope of smoothed BC lies on both accelerated BC parabolas, and thus is the intersection point of the two.

We consider only one parabola, as the case for the other one is similar. Without loss of generality, take its vertex to be $(0, 0)$, the point where the smoothing parabola (of second derivative $-s_l$) joins the BC parabola (of second derivative s_u) to be (a, b), and the maximum of the smoothing section to be (c, d).

Since (a, b) lies on the BC parabola, we have $b = \frac{1}{2}s_u a^2$. Since the smoothing parabola and the main parabola are tangent at (a, b), their gradients must be the same at (a, b). Thus $c = (s_l + s_u)a/s_l$. Since (c, d) lies on the smoothing parabola, $d = \frac{1}{2}s_l(c - a)^2 + \frac{1}{2}s_u a^2$. These simplify to give $d = \frac{1}{2}\frac{s_l s_u}{s_l + s_u}c^2$. Thus (c, d) lies on the general accelerated BC parabola as required.

4. The equivalence of one and two point methods

The result of the previous section where an aesthetically nice two point method is equivalent to a one point method holds for many algorithms. Our technique is to create a one point method which has the same peak points as the two point method. We note that the peak points are the only section of the upper envelope relevant to the performance of the algorithm.

We fix a point (x_0, y_0) and examine the peak points of the upper envelopes between (x_0, y_0) and arbitrary point (x_1, y_1). These points must be on the curve of our one point method. This procedure is illustrated in figure 7, where our one-point method must pass through all circled peak points.

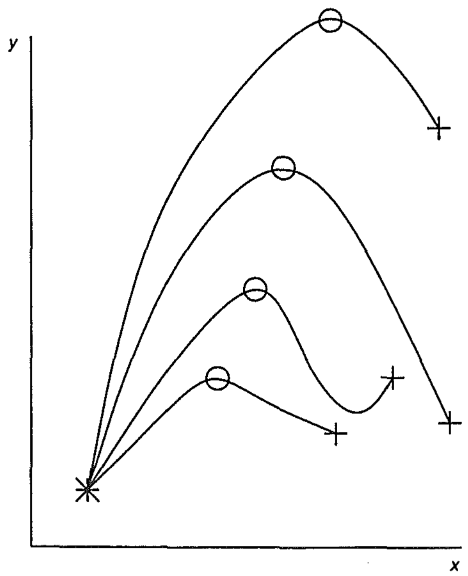

Figure 7. Some (x_1, y_1) [+] and their peak points [o]

Some technical difficulties are encountered when we attempt to formalise this. We note that it is usually only possible to form a one point method when the points produced lie on the graph of a function. To avoid the problems we required some technical assumptions, which we label as consistency conditions.

Consistency Condition 1 *Given a function f in the allowable class of functions and three points $(x_1, f(x_1)), (x_2, f(x_2)), (x_3, f(x_3))$ then if $x_1 < x_2 < x_3$ the maximum of the upper envelope on $[x_1, x_2]$ is less than or equal to the maximum of the upper envelope on $[x_1, x_3]$, as is the maximum of the upper envelope on $[x_2, x_3]$.*

This means that more distant points cannot be used to further improve the envelope between two adjacent points. This eliminates most pathological algorithms, but the difficulties with the end-points can still arise. The following condition eliminates most of these.

Consistency Condition 2 *Given (x_1, y_1) , (x_2, y_2) such that $x_1 < x_2$, $y_1 < y_2$ and the upper envelope on $[x_1, x_2]$ has no maximum on the interior of the interval, if we have any (x_3, y_3) such that $x_2 < x_3$, then if the maximum of the upper envelope* *on* $[x_1, x_3]$ *is in* $[x_1, x_2]$*, it is less than* y_2*. Similarly given* (x_1, y_1)*,* (x_2, y_2) *such that* $x_1 < x_2$*,* $y_1 > y_2$ *, and the upper envelope on* $[x_1, x_2]$ *has no maximum on the interior of the interval, if we have any* (x_0, y_0) *such that* $x_0 < x_1$*, then if the maximum of the upper envelope on* $[x_1, x_3]$ *is in* $[x_1, x_2]$*, it is less than* y_1*.*

This means that if a possible upper envelope on an interval has its maximum point at one of the end-points, no other upper envelope can have a higher maximum point in that interval. In some respects this is requiring the algorithm to be efficient, as the global optimum is known not to lie in that interval, so it would not be efficient to create another peak point in it. The theorem follows easily:

Theorem 2 *Given a two point algorithm satisfying both consistency conditions, there is an equivalent one point method.*

Note that the theorem only determines the shape of the one point method over some part of the domain. Often there are gaps as there may be regions where the peak point can not lie. Completions are non-unique, but this does not affect the operation of the algorithm.

Two applications of theorem 2 are theorem 1 and the result in [1], where a simplified form of the two-point method developed by Brent in [5] was shown to be equivalent to the one-point method developed in [1].

This is an example of a special case where the two-point method involves the placing of translations of a fixed function which we call a template to produce the upper envelope. This is an upside-down parabola in the case of Brent, and an inverted absolute value function in the case of Piyavskii/Shubert. We can show the following theorem which is illustrated in figure 8.

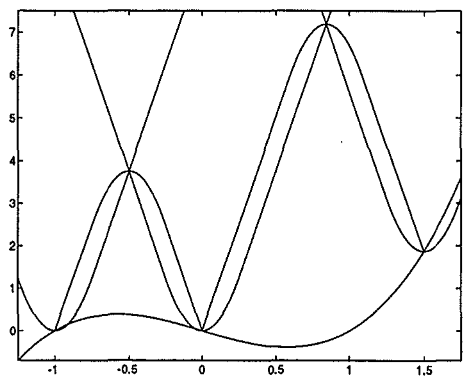

Figure 8. Unimodal template $\phi(x)$ and the corresponding cutter $-\phi(-x)$

Theorem 3 *Given a two-point method that uses a unimodal template $\phi(x)$ and satisfies the first consistency condition, a one point method obtainable by theorem 2 uses the function $-\phi(-x)$ as a cutter.*

This theorem is closely related to that in [1], where it was shown that if $\phi(x)$ lay below the graph of f at all global optima, then $-\phi(-x)$ could be used as a cutter. The Piyavskii/Shubert algorithm can also be regarded as a trivial example of theorem 3. The two-point view is of 'Λ-hats' being placed between adjacent evaluated points, whereas the one point view is of 'V-shapes' being placed at evaluated points.

5. Optimality for envelopes

The use of envelopes in the class of functions to be optimized is more than for aesthetic or 'closeness' reasons. In section 2.4 it was noted that smoothed BC used such envelopes. Because of this, it can be shown to be an optimal algorithm. One way to show that an algorithm in our general framework is optimal requires the upper envelope to be optimal in the following sense:

Definition. An upper envelope g is optimal if for every upper envelope h, $g(x) \geq h(x) \;\; \forall x \in [a, b]$.

The following theorem gives a sufficient condition for optimality of upper envelopes and implies that the envelopes constructed by the smoothed Breiman and Cutler algorithm are optimal.

Theorem 4 *If an upper envelope of f after n iterations, h_n, is a function in the class then it is optimal.*

Proof: Clearly, if C_n is the set of functions in the class for which the algorithm is designed have the same values on the n th evaluated set, then $g_n : [a, b] \to \mathbb{R}$ given by $g_n(x) = \sup_{f \in C_n} f(x)$ is the optimal upper envelope at the n th stage. Any upper envelope must lie above or on any function in the class, and thus above or on g_n. Thus $g_n \leq h_n$. But g_n is the pointwise supremum of functions in the class, so must lie above or on any function in the class. Thus $h_n = g_n$, and h_n is thus optimal.

Note that g_n is not necessarily a function in the class. For example consider differentiable functions with $|f'(x)| \leq m$. The upper envelope built up in this case is the same as for the class of Lipschitz functions with constant m, which it is a member of. It is not differentiable at the peak points, so not a function in the original class. Similarly we find that the upper envelope for smoothed BC is not a C^2 function, but it is in the class of functions with Lipschitz first derivatives and also an upper envelope for this class. This shows the converse of the theorem is false.

References

1. W. Baritompa. Customizing methods for global optimization — a geometric viewpoint. *J. Global Optimization*, 3:193–212, 1993.
2. W. Baritompa. Accelerations for a variety of global optimization methods. *J. Global Optimization*, 4:37–45, 1994.
3. W. Baritompa and A. Cutler. Accelerations for global optimization methods that use second derivative information. *J. Global Optimization*, 4:329–341, 1994.
4. Leo Breiman and Adele Cutler. A deterministic algorithm for global optimization. *Math. Program.*, 58:179–199, 1993.
5. Richard P. Brent. *Algorithms for minimization without derivatives.* Prentice-Hall, Englewood Cliffs, N.J., 1973.
6. J. Pinter. Extended univariate algorithms for n-dimensional global optimization. *Computing*, 36:91–103, 1986.
7. S.A. Piyavskii. An algorithm for finding the absolute extrema of a function. *USSR Comp. Math. and Math. Phys.*, 12:57–67, 1972.
8. Yaroslav D. Sergeyev. Global optimization algorithms using smooth auxiliary functions. University of Calabria, Department of Mathematics, Report n. 5, 1994.
9. Bruno O. Shubert. A sequential method seeking the global maximum of a function. *SIAM J. Numer. Anal.*, 9:379–388, 1972.