CUSTOMIZING METHODS FOR GLOBAL OPTIMIZATION - A GEOMETRIC VIEWPOINT

by

W.P. Baritompa
University of Canterbury, Christchurch, New Zealand

No. 80 March, 1993

Abstract – A new class of global optimization algorithms, extending the multidimensional bisection method of Wood, is described geometrically. New results show how the geometry of the global minimum relates to performance. Remarkably, the epigraph of the objective function, turned upside down, plays a key role. Algorithms customized to take advantage of special information about the objective function belong to the class. A number of algorithms in the literature, including those of Piyavskii-Shubert, Mladineo, Wood and Breiman & Cutler, also belong, and simple modifications of them produce customized algorithms. Comparison of various algorithms in the class is provided.

Keywords – Multidimensional bisection, deterministic, global optimization, mathematical programming

Customizing Methods for Global Optimization — A Geometric Viewpoint

W. Baritompa

Department of Mathematics
University of Canterbury
Christchurch, New Zealand

A new class of global optimization algorithms, extending the multidimensional bisection method of Wood, is described geometrically. New results show how the geometry of the global minimum relates to performance. Remarkably, the epigraph of the objective function, turned upside down, plays a key role. Algorithms customized to take advantage of special information about the objective function belong to the class. A number of algorithms in the literature, including those of Piyavskii-Shubert, Mladineo, Wood and Breiman & Cutler, also belong, and simple modifications of them produce customized algorithms. Comparison of various algorithms in the class is provided.

Keywords — Multidimensional bisection, deterministic, global optimization, mathematical programming

AMS 1991 Subject Classification 90C30, 65K05

1. Preliminaries

Introduction

The key contribution of this paper is that the shape of the epigraph near the global minimum plays an important role in the understanding of a new class of global optimization algorithms. These algorithms are geometric extensions of Wood's multidimensional bisection. For algorithms in this class, best performance comes from those that best incorporate the geometry of the global minimum. This means it is possible to find an algorithm in this class customized for objective functions with specific geometry at their global minimum.

The well known Piyavskii-Shubert algorithm is a very simple example of one of these extensions and can be used as an illustration. This algorithm requires, as a parameter, an upper bound for the Lipschitz constant of the objective function. This bound translates to a geometric fact about slopes. If additional geometry about the global minimum is known, namely that slopes around the global minimum are much smaller than this bound, this paper shows that running the algorithm with a parameter smaller than the

Lipschitz constant may give better performance. By using such a non-standard parameter, the algorithm is no longer Piyavskii-Shubert, but can be viewed as modification customized to use the additional information about the global minimum.

Outline

In the interest of completeness, this paper begins with the background context of multidimensional bisection. This is followed with an informal pictorial excursion motivating the formal results. Section 2 defines geometric extensions of multidimensional bisection, and shows many algorithms relying on underestimators or lower envelopes are such extensions. Section 3 contains the main result which implies non-trivial extensions. It formalizes customizing an algorithm to incorporate the geometry of the global minimum. Finally it offers some insight into the behavior of some algorithms when an incorrect Lipschitz bound is used. Section 4 describes classes of functions suited to customized methods. Section 5 discusses implementation. Section 6 gives computer tests which empirically verify that customized algorithms work better. Section 7 concludes with a summary and questions for future work.

Background

Wood [9,10] presents a multidimensional bisection algorithm for finding the global minimum of a Lipschitz continuous function defined on a compact domain in Euclidean space. As he points out, the most familiar "bisection" algorithm is that used to find the roots of a function of one variable by successive halving of an interval where the function changes sign. The salient feature of the root finding algorithm is that it starts with an initial bracketing interval which is successively divided into two parts, one of which contains the point of interest and so provides a better bracket.

In the one variable case, a brief geometric description of one variation of Wood's algorithm ("multidimensional bisection with complete reduction") is given here. Note in this one dimensional situation, the method reduces to the familiar Piyavskii-Shubert algorithm.

Let f be a Lipschitz continuous function of one variable and M be a bound for the Lipschitz constant. Multidimensional bisection produces a nested family of sets $B_0, B_1, \ldots$ (called the *brackets*) each containing the global minimum point(s) on the graph of f.

- Initial step: Let B_0 be a bracketing set consisting of a triangle containing the global minimum. Let $i = 0$
- Iterative step:

  (a) Increment i.
  (b) Let x_i be the first coordinate of the lowest point of the bracket. Compute $y_i = f(x_i)$.
  (c) (*Cutting*) Let B' be found by removing from B_{i-1} the region in the plane strictly below the downward facing cone (with sides of slope $\pm M$) with vertex at (x_i, y_i).
  (d) (*Capping*) Let B_i be found by removing from B' the half-plane above the line $y = y_i$.

- Stopping rule: If the vertical height of B_i is small enough, stop.

Figure 1 shows one iterative step when $M = 1$. The bracket prior to the function evaluation (the three triangles in lightly ruled lines) is changed to the improved bracket (the three triangles in darkly ruled lines).

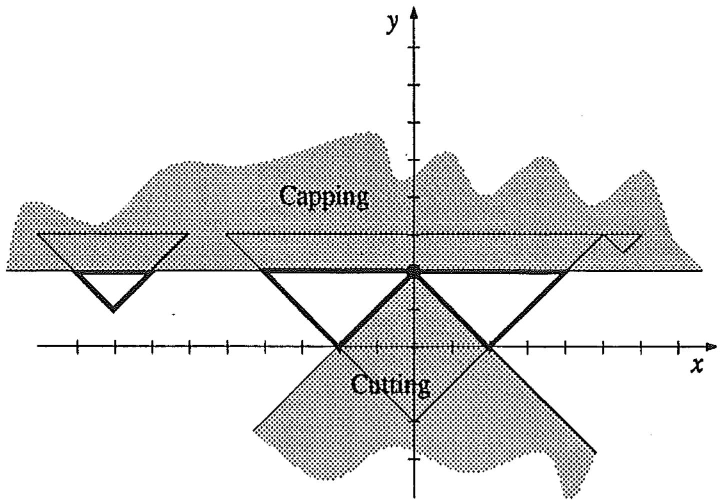


Figure 1 An iterative step of multidimensional bisection

Note "bisection" is a reasonable term to apply to this algorithm as all the salient features are present. At each iteration the plane is broken into two regions; a part (shaded) that cannot contain the global minimum and a part (unshaded) that does. This breaks up the previous bracket (here a finite union of triangles) into two parts, the correct part being kept for the new bracket.

For the purposes of this paper, all that is necessary is an understanding of these geometric ideas associated with multidimensional bisection. Namely how *cutting* and *capping* use new function evaluations to modify an old bracket of the global minimum, to produce a better one.

The important discovery made by Wood in [9,10] was an appropriate extension to higher dimensions. There the brackets consist of a finite union of truncated upward facing simplicial cones. Cutting consists of removing downward facing simplicial cones, while capping removes regions above hyperplanes through the evaluated points. The implementation of these geometric ideas is presented in detail in his papers. Additionally results about convergence, acceleration and optimality are given.

Motivation

The following pictorial excursion in Figures 2 to 5 motivates this paper's formal ideas. Suppose one is given an unknown Lipschitz continuous function with Lipschitz constant $M = 2$. Given some function values, consider trying to find a bracket for the global minimum.

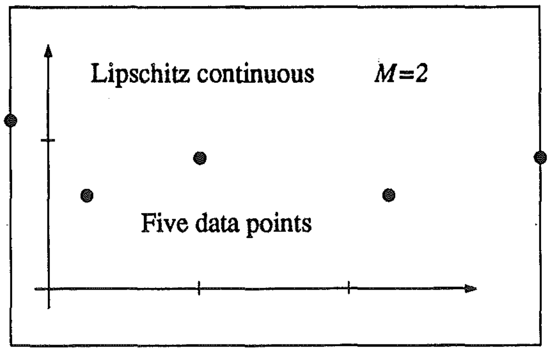


Figure 2 Some data for an unknown function

The geometric ideas of cutting and capping leave the following bracket (the unshaded region) which must contain the global minimum.

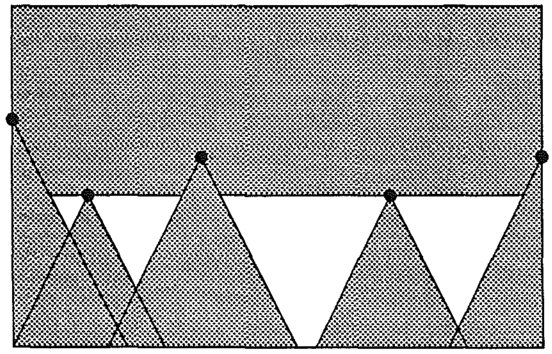


Figure 3 Bracket for global minimum when cutting away downward cones with sides of slope ± 2

Capping to the lowest value did not remove the global minimum because it has a lower value. Cutting away the cones did not remove it, because the cones lie under the graph. In fact cutting away the cones cannot remove any of the graph. Since the problem is just to find the global minimum, it seems plausible that cutting could be done with regions larger than cones.

With more specific information about the function, the goal is to find better cutting regions. Figure 4 shows two functions with Lipschitz constant $M = 2$ that fit the data. Their global minima belong to the bracket. Observe, for each, the graph is above the cones cut away.

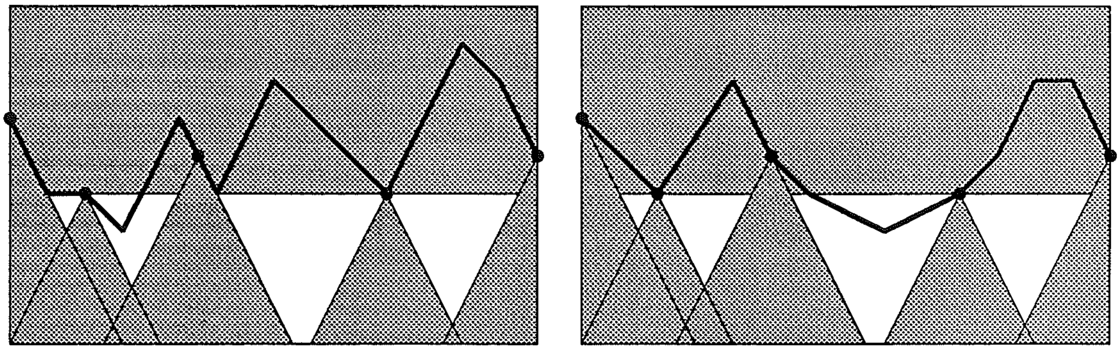
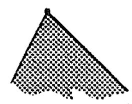
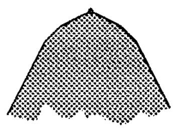


Figure 4 Two functions in relation to previous bracket

It is possible to use bigger cutting regions suited to each. An asymetric cone  suits the first, and a "barn roof"  suits the second, respectively. Figure 5 shows the better brackets that are produced.

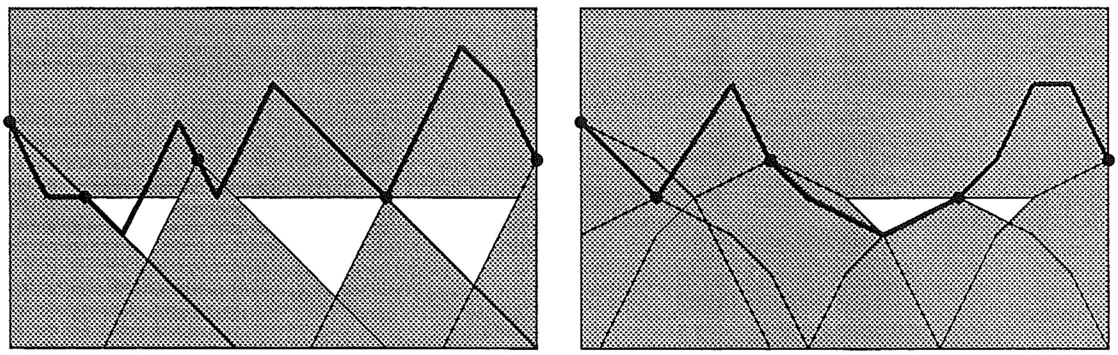


Figure 5 The two functions and better brackets found by using customized cutting regions

The willingness to cut away some of the graph produced smaller brackets. Perhaps the southern hemisphere viewpoint inspired this observation: Good cutting regions come from turning the epigraph of the function at the global minimum upside down.

The remainder of this paper formalizes this observation, shows how to customize a variety of global minimization algorithms, and provides a geometric explanation of aspects of the Piyavskii-Shubert algorithm and its generalizations.

Notation and basic problem

The basic problem is to find the global minimum α and its location $E = f^{-1}(\alpha)$ of a continuous function $f : K \to \mathbb{R}$ where K is a compact domain in $\mathbb{R}^n$. The global minimum can also be thought of as $G = \{(x,\alpha)|\ x \in K \text{ and } f(x) = \alpha\}$ a subset of $\{(x,y)|\ x \in K \text{ and } y \geq f(x)\}$, the *epigraph of f* in $\mathbb{R}^{n+1}$. The sample sequence of points where the function has been evaluated is denoted $\{x_i\}$, and the lowest known height is $\alpha_i = \min_{j \leq i} f(x_j)$. Given $u : \mathbb{R}^n \to \mathbb{R}$, saying u is an *overestimator (or upper envelope) of f over K* means $\forall_{x \in K}\ \ u(x) \geq f(x)$. *Underestimator (lower envelope)* is defined analogously.

The *upward M−cone* is $\{(x,y)|y \geq M\|x\|\}$ in $\mathbb{R}^{n+1}$. The *upward simplicial M−cone* is a cone over the regular n−simplex inscribed in the above. The *upward B−paraboloid* is $\left\{(x,y)\middle|y \geq \frac{1}{2}B\|x\|^2\right\}$. The *upward MB−parabolic cone* is the union of the upward B−paraboloid and $\left\{(x,y)\middle|y \geq \max\left\{\frac{M^2}{2B}, M\|x\| - \frac{M^2}{2B}\right\}\right\}$, the part of the circumscribed translated M−cone above the level of tangency. Note these regions are just epigraphs of quite simple functions on $\mathbb{R}^n$. Often more general regions C in $\mathbb{R}^{n+1}$ need to be considered. The *vertex* of all the above regions is considered to be the origin. A region C *turned upside down* is formally $-C$. To say a function is bounded by one of the above at a point x_0 in the domain means that the region with its vertex translated to $(x_0, f(x_0))$ lies in the epigraph of f. For those readers more familiar with a non-geometric viewpoint, saying a function f is bounded by an upward M−cone at x_0 is the same as saying $g(x) = f(x_0) + M\|x - x_0\|$ is an overestimator of f over K.

The downward versions of the above are defined appropriately. In particular being bounded by lower regions corresponds to underestimators.

Using these notions, the class of Lipschitz continuous functions $L(M)$ consists precisely of those functions which are bounded by upward and downward M−cones at each point of the domain. Similarly let $LS(M)$ be the class of functions bounded above and below by simplicial M−cones at each point of the domain.

2. The Geometric Viewpoint

Geometric Extensions of Multidimensional Bisection (GEMB)

One obvious way to generalize multidimensional bisection is to use different regions in place of $M-$cones at the cutting step (c). Additionally the strategy at step (b) of choosing the next point for function evaluation can be generalized, but this is of minor importance in this paper. Except for the later results in section 3, the choice of first coordinate of the lowest point of the bracket suffices.

Concentrating on the first aspect yields the following family of geometric extensions of multidimensional bisection. Each cutting strategy (step (c)) produces a different algorithm. Here the brackets $B_0, B_1, \ldots$ are in $\mathbb{R}^{n+1}$.

- Initial step: Let B_0 be a bracketing set containing the global minimum. Let $i = 0$
- Iterative step:
  - (a) Increment i.
  - (b) Choose x_i. Compute $y_i = f(x_i)$.
  - (c) (*Cutting*) Let B' be found by removing from B_{i-1} a cutting region at (x_i, y_i).
  - (d) (*Capping*) Let B_i be found by removing from B' the region above the hyperplane $y = y_i$.
- Stopping rule: If the vertical height of B_i is small enough, stop.

Of course an extension of multidimensional bisection is only of interest if the brackets always contain G. Results pertaining to this are given in sections 3, 4 and 5.

Examples of Geometric Extensions Of Multidimensional Bisection

A number of algorithms in the literature are GEMB. Of course Wood's multidimensional bisection is an example where the cutting regions are downward simplicial $M-$cones translated to the evaluated point on the graph. Similarly Mladineo's algorithm [6] uses translated downward $M-$cones. Breiman & Cutler's algorithm [5] uses downward $B-$paraboloids translated so they are tangent to the graph at the evaluation points.

The later two algorithms were not presented by their authors as geometric extensions of multidimensional bisection, but were described using underestimators. The global minimum of the underestimator being taken as the approximation to the global minimum of f. However such algorithms have a natural bracket associated with the underestimator consisting of all points above or equal to the underestimator and below or equal to the lowest known value. If the underestimator is the point-wise maximum of simple functions, using the regions below the graphs of these simple functions as cutting regions for GEMB produce algorithms that give these natural brackets. Thus the descriptions of these two algorithms as GEMB follow since Mladineo used an underestimator of the form $\max_{j\leq i} \{f(x_j) - M\|x - x_j\|\}$ and Breiman & Cutler used an underestimator of the form $\max_{j\leq i} \left\{f(x_j) + \nabla f(x_j)(x - x_j) - \frac{1}{2}B\|x - x_j\|^2\right\}$.

Some non-trivial GEMB do not correspond to using underestimators. As shown in the pictorial motivation, there are brackets that do not arise as the region above an underestimator and below the lowest known value. It is possible to have a bracket containing the global minimum but not all the graph below the lowest known value. In these cases GEMB use cutting regions which remove parts of the graph of f. The next section describes which cutting regions can be used in certain circumstances.

3. Main observation — Custom Cutting Regions

Getting Custom Cutters

In this section we show how geometric extensions of multidimensional bisection can be found that do not arise using underestimators. The main result in this section shows how specific information about the global minimum point being sought can be used to get a "template." The cutting regions are all found by translating this template to the point of evaluation. As well as providing for new GEMB, the main results provide some geometric insight to the behavior of certain algorithms.

Definition 3.1 *An <u>upper fitting for f and N over K</u> is a function* $u_0 : \mathbb{R}^n \to \mathbb{R}$ *such that for each* $(x_n, y_n) \in N$, $u(x) = y_n + u_0(x - x_n)$ *is an overestimator of* f *over* K.

Of particular interest is the case when $N = G$. Examples are: a function u_0 with epigraph the upward $M-$cone where M is a upper bound for the Lipschitz constant; u_0 with epigraph the upward B–paraboloid where B is an upper bound for the eigenvalues of the Hessian, and the global minimum of f occurs where the gradient is zero; and $u_0(x) =$

$\max\limits_{(x_m, f(x_m)) \in G} \{f(x + x_m) - f(x_m)\}$ for f defined on all of $\mathbb{R}^n$. In light of the next proposition and later tests in section 6, this last example produces GEMB most customized to the objective function. However it is not useful in practice as it requires complete knowledge of the objective function. It does provide a useful reference for comparing other GEMB. Note: for the examples here $u_0(0) = 0$, but this is not necessary.

Geometrically (Figure 6) consider a set C the epigraph of an upper fitting for f and G over K. This means that for any point $\mathbf{v} \in G$, $(\mathbf{v} + C) \cap (K \times \mathbb{R})$ belongs to the epigraph of f. The following proposition, which formalizes the southern hemisphere observation in the motivation section, shows $-C$ can be used as a template for the cutting regions.

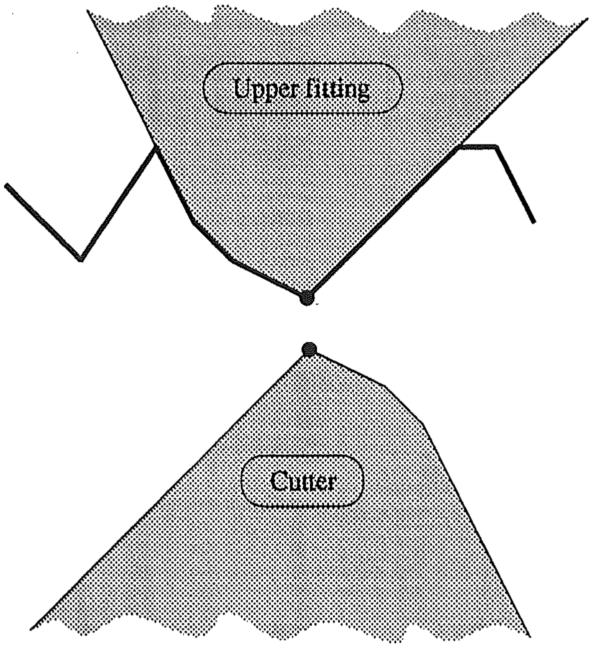


Figure 6 Turning a fitting upside down

Proposition 3.2 *If u_0 is an upper fitting for f and G over K, then for each function evaluation at $x_e \in K$, the region below the graph of $g(x) = f(x_e) - u_0(x_e - x)$ when used as a cutting region in geometric extensions of multidimensional bisection, will not remove any points of G.*

Proof: Take $x_e \in K$. Given any global minimum $(x_m, f(x_m)) \in G$, one needs to show it is on or above the graph of g. By assumption $u(x) = f(x_m) + u_0(x - x_m)$ is an overestimator of f over K so

$$u(x_e) \geq f(x_e)$$
$$f(x_m) + u_0(x_e - x_m) \geq f(x_e)$$
$$f(x_m) \geq f(x_e) - u_0(x_e - x_m) = g(x_m) \quad \blacksquare$$

Before looking at realization and performance of such GEMB, we observe that proposition 3.2 provides some geometric insight to the behavior of certain algorithms. To illustrate this consider using the Piyavskii-Shubert algorithm. It requires an estimate for the Lipschitz constant M. Suppose one mistakenly thought M was 1, when in fact it was really 2. In other words consider a case where a wrong "Lipschitz constant" is being used.

Three possibilities arise: first the algorithm may work correctly; second it may satisfy the stopping criterion but give a bracket that fails to contain the global minimum; and third it may produce an empty bracket at some step (i.e. in terms of underestimators, the one found at that step will lie completely above the lowest known value). The concept of upper fitting provides insight into these possibilities.

In the first situation, the Lipschitz constant $M = 1$ may not have been correct, but if $u_0(x) = |x|$ is an upper fitting for f and G over the domain, then proposition 3.2 insures that if downward $1-$cone are used, the global minimum will always be in the brackets.

The third possibility of getting an empty bracket is illuminated by the following results which consider what happens when various cutting regions are used in GEMB. It should be noted here that capping step (d) of GEMB, could be deferred. In other words given a sample sequence, a region F_k could be found by removing all cutting regions from $K \times \mathsf{R}$, the bracket B_k is then found by removing from F_k all points above the lowest known value. The following propositions concern the sets F_k. Consider GEMB that use a template $-C$ for cutting regions, where C is the epigraph of some function $u_0 : \mathsf{R}^n \to \mathsf{R}$ not necessarily an upper fitting for f and G over the domain. What such an algorithm will find is examined first.

Proposition 3.3 *Given a function $u_0 : \mathsf{R}^n \to \mathsf{R}$. Let N be the biggest set in R^{n+1} such that u_0 is an upper fitting for f and N over K.* $F_k = \left\{ (x_f, y_f) \middle| \; y_f \geq \max_{i=1\cdots k} \{ f(x_i) - u_0(x_i - x_f) \} \right\}$ *is the set of points outside all cutting regions after k function evaluations at a sample sequence $\{x_i\}$.* $F_\Omega = \{ (x_f, y_f) | \, \forall x \in K \;\; y_f \geq f(x) - u_0(x - x_f) \}$ *is the set of points outside all cutting regions if evaluations were done at all points of K. Let $F_\infty = \bigcap_{k=1}^{\infty} F_k$. The following holds* $F_k \supset F_{k+1} \supset \cdots \supset F_\infty \supset F_\Omega = N$.

Proof: Clearly N is $\{(x_n, y_n) | \, \forall x \in K \;\; y_n + u_0(x - x_n) \geq f(x)\}$ which equals F_Ω. ■

Note for the following discussion is it necessary to consider strategies at step (b) other than choosing the first coordinate of deepest point in the bracket. This next proposition gives sufficient conditions for the third possibility.

Proposition 3.4 *Consider a geometric extension of multidimensional bisection that uses an inappropriate template $-C$ for cutting regions, where C is the epigraph of some function $u_0 : \mathbb{R}^n \to \mathbb{R}$ with $u_0(0) = 0$, but u_0 is not an upper fitting for f and any set containing the global minimum over K. Furthermore if the sampling strategy always produces sample sequences that are dense in K, then the algorithm will eventually produce an empty bracket.*

Proof: Proposition 3.3 and the density of the sample sequence provides that $F_\infty = F_\Omega = N$. The condition $u_0(0) = 0$ implies that F_Ω is a subset of the epigraph of f, and hence lies on or above the hyperplane at height α. The fact that u_0 is not an upper fitting for f and any set containing the global minimum over K means the set F_Ω strictly above the hyperplane at height α. Compactness of the domain provides that for some index k, F_k is sufficiently close to N and α_k is sufficiently close to α. So the bracket B_k being the points in F_k that lie at height α_k or below is empty. ■

Illustrated in Figure 7 is the set N where $1-$cones are used inappropriately. If enough points are sampled, F_k will be very close to N and some sample points will have values completely below it.

Considering the second possibility where an algorithm fails by stopping with a bracket that does not contain the global minimum. Without denseness of the sample sequence not much can be said, however one still has F_Ω strictly above the hyperplane at height α. Certainly it may happen for some sample sequence, F_k gets close enough to F_Ω to be higher than α, while B_k is a non-empty set small enough to satisfy the stopping criterion.

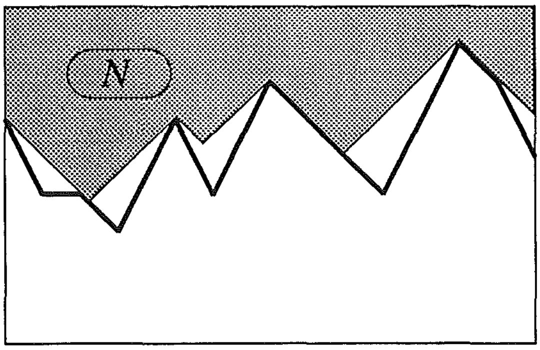


Figure 7 The set N when 1–cones are used inappropriately

The preceding results have analogs for capping. The concept of *lower fitting* is defined analogously. When looking for global minimum, one lower fitting that always works for any function is $l_0(x) = 0$. This gives rise to the capping removal region of the upper half-hyperplane. It is possible to further generalize multidimensional bisection by considering more interesting capping regions.

Global optimization via cutting and capping can be view as part of a more general framework. The bisection idea can be used to search for any point of any region in a vector space. It is possible to formalize this so the proposition 3.2 is a simple corollary of a more general result.

4. Classes of functions suited to custom cutting regions

Proposition 3.2 provides a condition for GEMB to perform correctly in the sense that the bracket at each stage contains G. Rates of convergence are handled empirically by examples provided in section 6 which support the observation that the use of bigger cutting regions generally produces a faster converging algorithm.

Performance has to be taken in the context of the class of objective functions with which the algorithm is designed to work. For example with a given objective function, an algorithm using both first and second derivative bounds would be expected to perform better than one that uses only first derivative information. For this reason a number of classes of functions is defined, and their relationships to each other are provided. Some classes have been used by previous authors, while others are specific to this paper and relate to the geometry of the global minimum.

Brent and Breiman & Cutler deal with functions with bounds on the second derivatives. Let $C^2_u(B)$ be the class of all twice differentiable functions such that $h(x_0 + \Delta x) = f(x_0) + \nabla f(x_0)\Delta x + \frac{1}{2}B\|\Delta x\|^2$ is an overestimator of f over K. Similarly let $C_l^2(\underline{B})$ have $h(x_0 + \Delta x) = f(x_0)+\nabla f(x_0)\Delta x-\frac{1}{2}\underline{B}\|\Delta x\|^2$ as an underestimator. For a given function the best bounds are the maximum and the negative of the minimum of the eigenvalues of the Hessian. Note the bounds B and $\underline{B}$ may be quite different.

Knowing that the global minimum occurs where the gradient is zero is quite useful. Let ZG be the set of differentiable functions with the global minima having zero gradient.

The following classes are ones suited to particular cutting regions arising from upper fittings. $SG(M)$ is the set of all functions with the upward simplicial M-cone at the global minimum contained in their epigraph. $CG(M)$ is the set of all functions with the upward M-cone at the global minimum contained in their epigraph. $PG(B)$ is the set of all functions with the upward B-paraboloid at the global minimum contained in their epigraph. $PCG(M, B)$ is the set of all functions with the upward $MB-$parabolic cone at the global minimum contained in their epigraph.

Proposition 4.1 *The following shows the relation between these classes:*

(1) $L(M) \subset LS(M)$

(2) $LS(M) \subset SG(M)$

(3) $L(M) \subset CG(M)$

(4) $C_u^2(B) \cap ZG \subset PG(B)$

(5) $L(M) \cap PG(B) \subset PCG(M, B)$

(6) $L(M) \cap C_u^2(B) \cap ZG \subset PCG(M, B)$

Proof: For (4) consider $f \in C_u^2(B) \cap ZG$ and a global minimum $(x_m, f(x_m))$ of f. $\nabla f(x_m) = 0$ so $h(x_m + \Delta x) = f(x_m) + \frac{1}{2}B\|\Delta x\|^2$ is an overestimator for f which means $f \in PG(B)$. For (5) consider $f \in L(M) \cap PG(B)$ and a global minimum $(x_m, f(x_m))$ of f. The $B-$paraboloid at the global minimum is in the epigraph of f and at each point of the epigraph the $M-$cone stays in the epigraph. The $MB-$parabolic cone is the union of all $M-$cones placed at points of the $B-$paraboloid. (6) follows from (4) and (5). ■

The purpose for introducing these classes of functions is to identify those aspects of the objective function that relate to various implementations of GEMB. However one may ask, is there any practical way to recognize to which class a function belongs? In some cases this may be possible, but generally this is a difficult problem. However the difficulty is not just restricted to the special classes introduced in this paper. For example finding M, such that a function belongs to $L(M)$, is itself a global optimization problem.

There can be a situation where it is possible to identify one of these classes without knowing G beforehand or having to find bounds at all points of the domain. For example, consider a function of one-variable known to have the global minimum at an interior point and defined by the differential equation $y' = x + g(y)$. Therefore $y'' = 1 + y'g'(y)$, and when $y' = 0$, $y'' = 1$. It follows that the function belongs to $PG(1)$.

5. Implementations of Geometric Extensions Of Multidimensional Bisection

Conceptually removing regions is easy, however, in practice can be difficult (e.g. it often requires finding the global minimum of another function and setting up of data structures to represent the geometry). For removal regions which are cones and paraboloids, implementations, done by others, are discussed below. More complicated regions can be handled in some cases. These are only briefly mentioned and are topics for other papers.

Existing methods

Interestingly enough, for the algorithms mentioned in this section, implementation is done by an appropriate choice of input parameters. The following two algorithms are implementations of GEMB with cone cutting regions. The multidimensional bisection algorithm of Wood was designed to require as input parameter a value M (ideally the smallest) such that the objective function is in $LS(M)$, however a smaller parameter will often do.

Remark 5.1 *For functions in SG(M), the multidimensional bisection algorithm can be used with parameter equal to* M.

The algorithm of Mladineo was designed to require as input parameter an upper bound for the Lipschitz constant. That is a value M (ideally the smallest) such that the objective function is in $L(M)$.

Remark 5.2 *For functions in CG(M), the algorithm of Mladineo can be used with parameter equal to M.*

Breiman and Cutler describe a method using the gradient function and bound $K = \underline{B}/2$, where the function belongs to $C_l^2(\underline{B})$. Since their algorithm deals with intersecting paraboloids, it implements GEMB with lower $B-$paraboloids as cutting regions.

Remark 5.3 *For functions in PG(B), the algorithm of Breiman & Cutler with the gradient function taken as constantly zero and $K = B/2$ can be used.*

In the one dimensional case, the essentials of the previous algorithm appeared in earlier literature. Brent produced an algorithm requiring an upper bound B on the second derivative. It relied on the fact (Theorem 2.1 in [4]) that the quadratic passing through the end points of an interval and having second derivative B, is an underestimator over the interval. A simplified version of Brent's algorithm is as follows: begin with evaluations at the end points of an initial interval; in general, form the piecewise quadratic with second derivative B between successive sample points; use the deepest point of the envelope for the next sample point.

Remark 5.4 *For functions of one variable in $C_u^2(B) \cap ZG$ the simplified version of Brent's algorithm above gives the same sample points as the algorithm of the previous remark.*

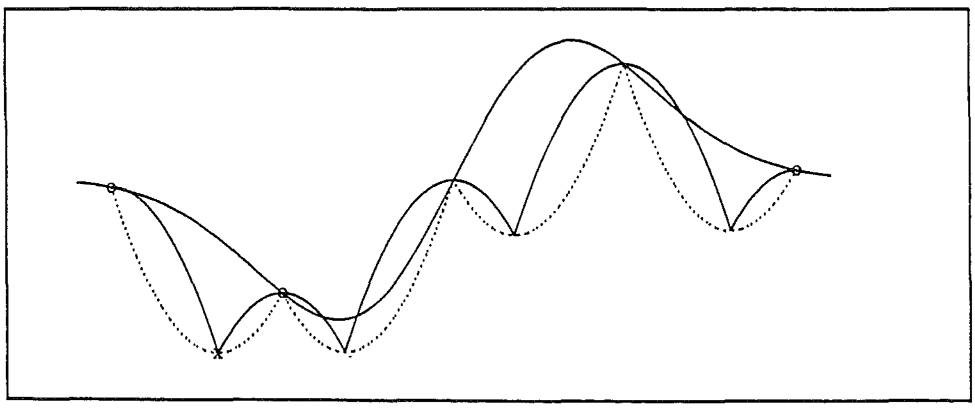


Figure 8 Brent's envelope vs. Parabolic envelope

Acceleration of existing methods

The previous remarks show existing algorithms implement some GEMB with no modification other than using different input parameters. With minor modification, these algorithms can be accelerated to handle cutting regions nearly as large as $MB-$parabolic cones. In particular the algorithms of Mladineo and Wood requiring a Lipschitz constant bound can be modified to use gradient calculations and second derivative bounds as well. The algorithm of Breiman and Cutler can be modified to use a Lipschitz constant bound. The required modifications and performance of the accelerations are examined in [1].

A Dual Implementation Of Multidimensional Bisection Which Implements GEMB In Special Cases.

In [2] an implementation of GEMB which handles cutting regions that are convex sets of the form $Ax \leq b$ is presented. The focus of that paper deals with the technicalities of implementing the geometry of multidimensional bisection via formulae dual to those used in Wood's papers. Each matrix A gives a different method. In particular cones over any polyhedron can be used, thus providing a spectrum of GEMB. At one end is Wood's multidimensional bisection with cutting regions cones over the simplex, the simplest polyhedron. At the other end is Mladineo's algorithm using cones over the sphere, the limiting "polyhedron."

6. Examples

Discrete Testing

In order to compare various GEMB a discrete implementation has been written in `matlab`. This implementation finds the global minimum of the objective function over $K_D = K \cap D$ where D is a finite discrete set. The version using cutting regions coming from an upper fitting u_0 is:

- Initial step: Let $e_0(x) = -\infty$ for each point $x \in K_D$. Let $\alpha_0 = \infty$. Let x_0 be a specified point of D to be used as the starting point. Let $i = 0$
- Iterative step:

  (a) Compute $y_i = f(x_i)$.
  (b) Increment i.
  (c) (*Cutting*) Let $e_i(x) = \max\{e_{i-1}(x), y_{i-1} - u_0(x_{i-1} - x)\}$.
  (d) (*Capping*) Let $\alpha_i = \min\{y_{i-1}, \alpha_{i-1}\}$.

(e) Choose x_i to be a point in K_D that minimizes $e_i(x)$.

- Stopping rule: If $\alpha_i - e_i(x_i)$ is small enough, stop.

The functions $e_i(x)$ play a similar role to underestimators. However it should be stressed that usually they are not underestimators, but they do have the property that their global minimum lie below G. The fact that K_D is finite means $e_i(x)$ can be kept as a finite list and step (e) is found by sorting.

The question of the correctness of this discrete implementation arises. Note if K is used in place of K_D one gets a description of GEMB for the original problem, however step (e) is no longer trivial and depends specifically on the complexity of u_0. Although not specifically needed, the bracket B_i is $\{(x, y) | \; x \in K_D, \; e_i(x) \leq y \leq \alpha_i\}$. The points of $K_D \times \mathbb{R}$ that would be in the brackets found by an exact implementation using the same sample path as the discrete version are precisely these B_i. Empirically, discrete versions of Wood's and Mladineo's algorithms have been compared with exact versions for functions of one and two variables. Convergence behavior is similar until the accuracy approaches the grid size, when the discrete version then finds the correct grid point very quickly.

Test Set Up

Five or six versions of GEMB were tried on four standard test functions which have their global minimum interior to their domains. The discrete implementation was used with D consisting of a 201×201 regular grid. The domain and starting points for these functions is as in [5]. For each the smallest constants (see Table 3) are found so the function belongs to $CG(M')$, $L(M)$, $C_u^2(B)$, and $C_l^2(\underline{B})$. Thus M is the Lipschitz constant, while M' is smaller. Particulars of the GEMB appear in table 1. The GEMB II, III, V, VI are algorithms customized to the objective function. The cutting regions they use come from regions that fit very well into the epigraph of the objective function at the global minimum.

GEMB	Cutting region used	Upper fitting $u_0(x) =$	Algorithm
I	$M-$cones	$M\|x\|$	Mladineo's
II	$MB-$parabolic cones	$\begin{cases} \frac{1}{2}B\|x\|^2, & \|x\| \leq \frac{M}{B} \\ M\|x\| - \frac{1}{2}\frac{M^2}{B}, & \|x\| \geq \frac{M}{B} \end{cases}$	
III	$B-$paraboloids	$\frac{1}{2}B\|x\|^2$	As in remark 5.3
IV	tangent $\underline{B}-$paraboloids	– not appropriate –	Breiman and Cutler's
V	M'–cones	$M'\|x\|$	As in remark 5.2
VI	From epigraph	$\max_{(x_m, f(x_m)) \in G} \{f(x + x_m) - f(x_m)\}$	Remarks before Prop. 3.2

Table 1 GEMB used in tests

A very simple indication of performance is the number of iterations till the bracket consists of one grid point. Although this is dependent on the initial grid size, it is useful for comparison. Note that an iteration of Breiman and Cutler's method (IV) consists of both a function and a gradient evaluation, while for the rest an iteration is a function evaluation. Table 2 summarizes these results (note: iterations exceeding a predetermined limit are indicated with a $>$ sign).

Test	I	II	III	IV	V	VI
-EXP2	>30	8	10	27	10	6
-COS2	>100	56	57	68	58	6
-RCOS	>300	242	247	221	—	13
-C6	>300	>300	>300	104	—	5

Table 2 Iterations till one grid point in bracket

Test	Domain	Initial Point	M	M'	B	$\underline{B}$
-EXP2	$(-1,1) \times (-1,1)$	$(0.2, 0.2)$	0.61	0.45	1	0.37
-COS2	$(-1,1) \times (-1,1)$	$(0.5, 0.5)$	4.8	1.9	26.7	22.7
-RCOS	$(-5,10) \times (0,15)$	$(0,5)$	113.6	—	29.2	16.8
-C6	$(-5,5) \times (-5,5)$	$(0,0)$	5601	—	5628	8.94

Table 3 Parameters used

More detailed convergence behavior can be seen in Figures 9–12 where the logarithm of the variation (i.e. the height of the bracket) is plotted against the iteration number.

The tests empirically verify that the use of bigger cutting regions give better performance. Using the epigraph (VI) itself provides the most customization for the given f and gives the best performance. Of course it is not a practical algorithm at all, but is an useful benchmark. The curves when $MB-$parabolic cones (II) were used generally are uniformly better than when either $M-$cones (I) or $B-$paraboloids (III) were used. For cones, using M' (V) is much better than using the Lipschitz constant. M (I).

A few specific comments can be made. Figures 9 and 10 use functions with a small Lipschitz constant relative to the second derivative bounds. Performance of the two versions using paraboloids (III and IV) is similar (hence GEMB using $B-$paraboloids is better as no gradient is needed) until the final iterations, where using the $B-$paraboloid converges very quickly. This is mostly due to the fact that for these functions the upward $B-$paraboloid is a very good local approximation at the global minimum. In Figures 11 and 12, the Lipschitz constant is large, and the bound $\underline{B}$ is substantially smaller than B. Here the method of Breiman and Cutler (IV) outperforms the other method using paraboloids (III). This is especially apparent in Figure 12.

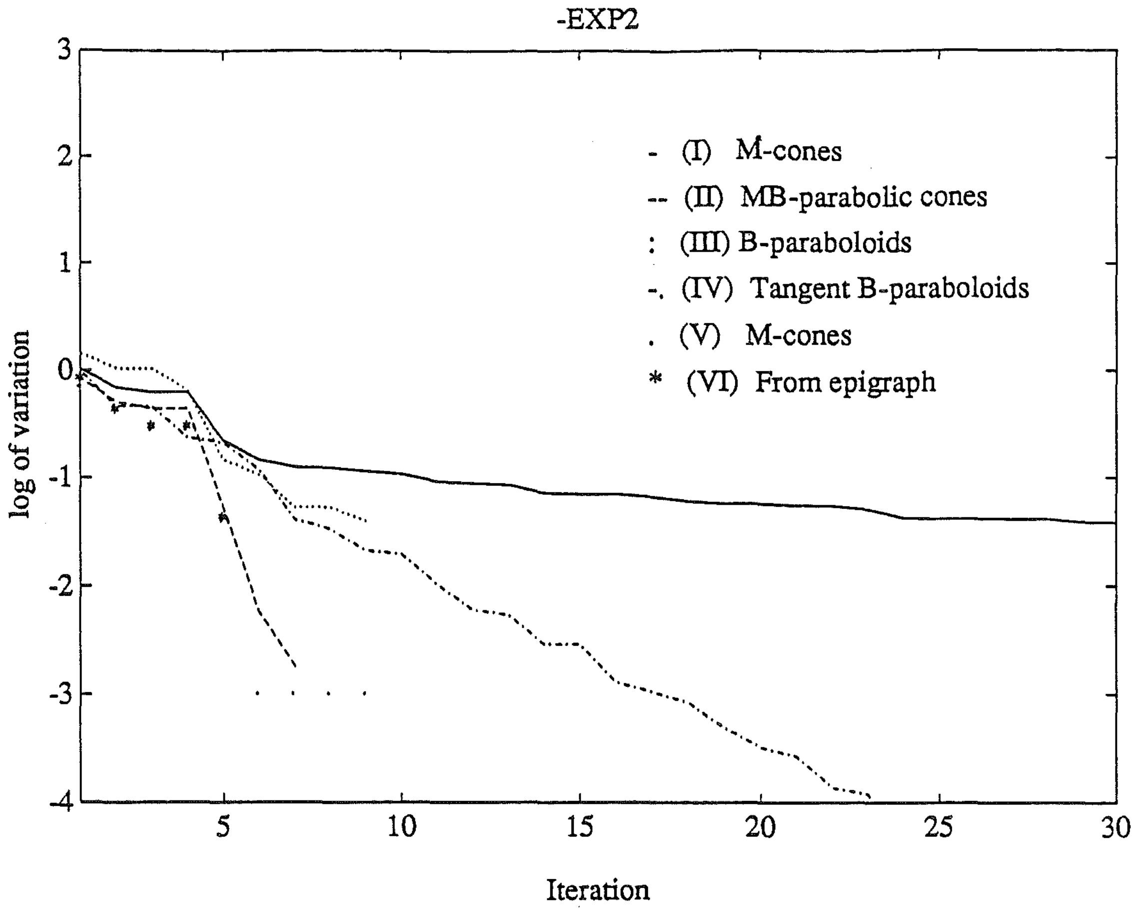


Figure 9 Exponential $f(x) = -e^{-\frac{(x^2+y^2)}{2}}$

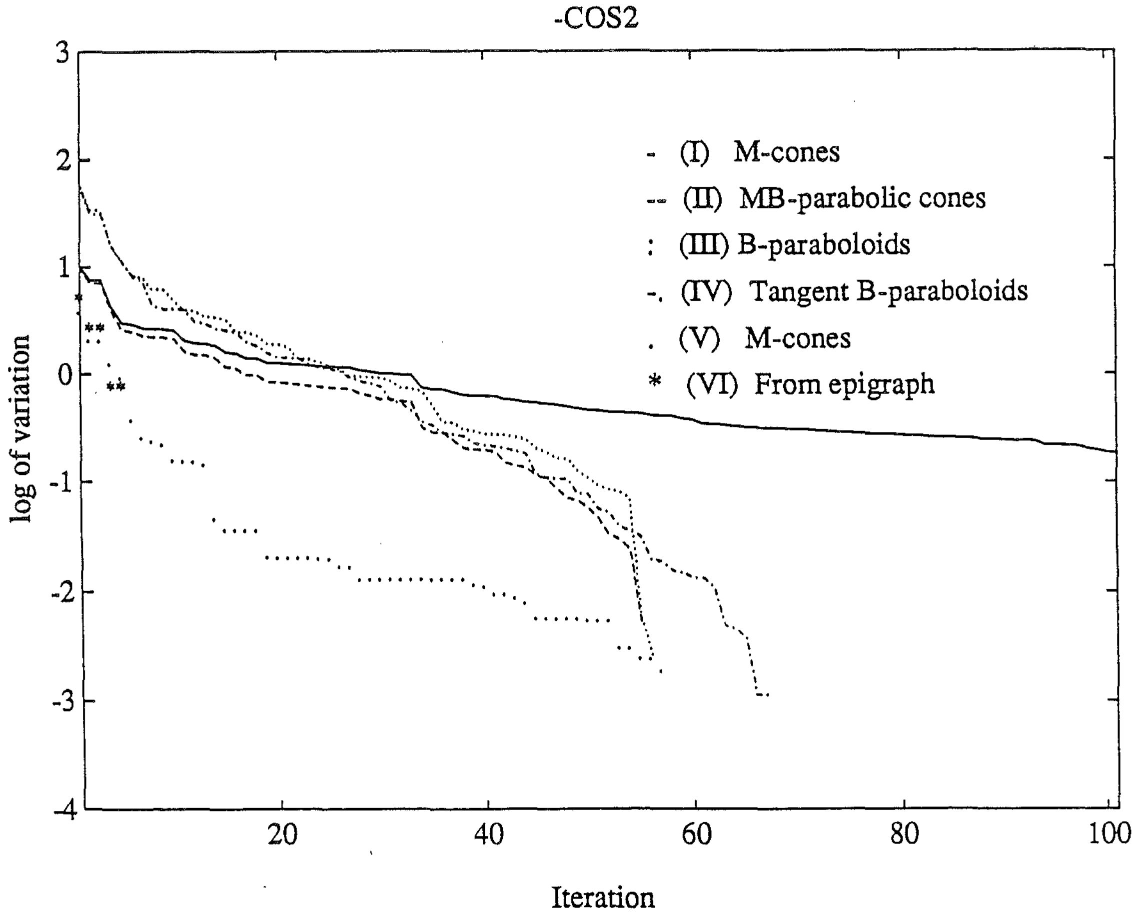


Figure 10 Cosine Mixture $f(x) = -\frac{1}{10}\cos 5\pi x - \frac{1}{10}\cos 5\pi y + x^2 + y^2$

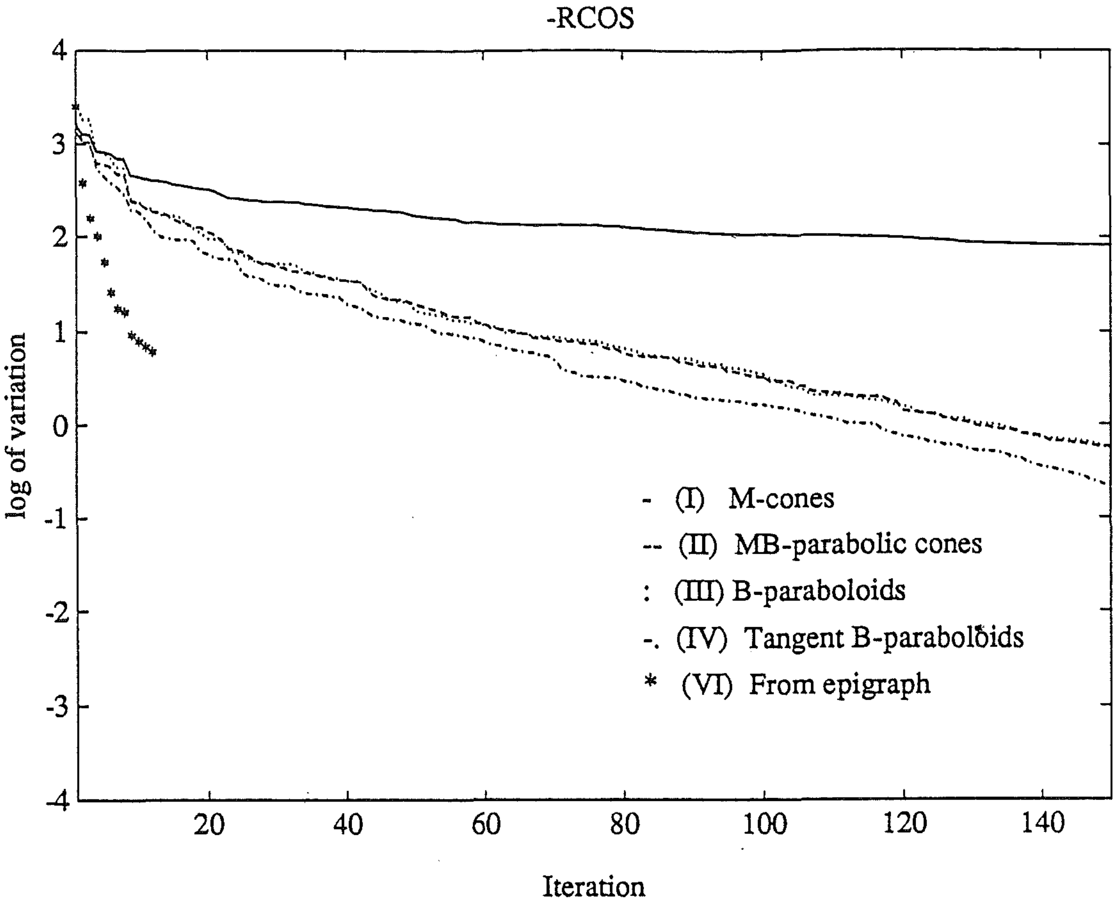


Figure 11 Branin $f(x) = \left(y - \frac{5.1}{4\pi^2}x^2 + \frac{5}{\pi}x - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x + 10$

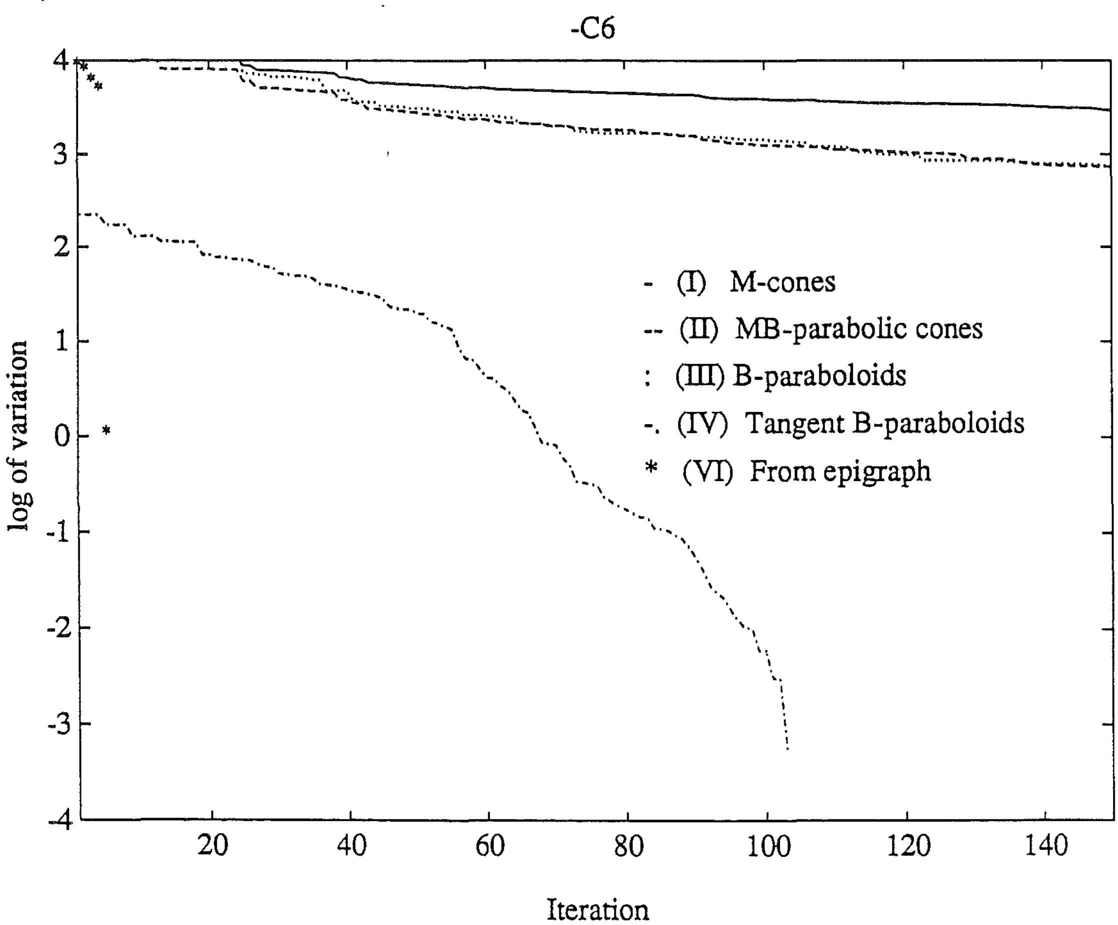


Figure 12 Six-Hump Camel Back $f(x) = 4x^2 - 2.1x^4 + \frac{1}{3}x^6 + xy - 4y^2 + 4y^4$

7. Summary and Future Directions

Put in the context of a geometrical extension of multidimensional bisection, the relationship of the geometry of the global minimum to the cutting regions has been shown to be a key factor guaranteeing brackets contain global minimum. Empirical tests verify that the closer the cutting region conforms to the epigraph at the global minimum, the better the convergence rates.

The algorithms of Breiman & Cutler, Mladineo and Wood with appropriate choice of input parameters not originally envisioned were shown to be implementations in this context. These and other implementations still handle relatively simple cutting regions. Implementing GEMB using more interesting cutting regions is an interesting area for future work.

A simple example was given in section 4 that showed customized algorithms might be practical for the class of solutions to a differential equation. This was because facts about the geometry of the global minimum could be deduced without knowledge of its location or value. What are some other practical classes of functions? Under what circumstances can the geometry of the global minimum be practically deduced?

Another area for future work concerns other sampling strategies. Sometimes sample points far away from the global minimum, but with large function values, cause a large area to be cut away. This saves the algorithm work later on. Is it worth trying to evaluate at points with large values in hope of cutting away large regions? Can this be done by trying to find the global minimum and maximum simultaneously? Next point strategies that choose evaluation points randomly may be more effective.

Acknowledgments

The author wishes to thank the International Institute for Applied Systems Analysis for invitation to the II Workshop on Global Optimization. Thanks to my colleague Graham Wood for sharing his work on multidimensional bisection. The enthusiasm and encouragement of the optimization team of Graham Wood and Zhang Baoping has been a major stimulus.

References

[1] W. Baritompa (1990), Accelerating Methods for Global Optimization, Research Report, University of Canterbury, Christchurch, New Zealand

[2] W. Baritompa (1990), A dual view of multidimensional bisection —Extensions and Implementation, Research Report, University of Canterbury, Christchurch, New Zealand

[3] Zhang Baoping, W. Baritompa and G. R. Wood (1990), Multidimensional Bisection: the performance and the context, preprint

[4] R. P. Brent (1973), Algorithms for Minimization without Derivatives, Prentice Hall 81–115

[5] Breiman & Cutler(1989), A Deterministic Algorithm for Global Optimization, *Math. Program..* To appear

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

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

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

[9] G. R. Wood (1992), The bisection method in higher dimensions, *Math. Program..* To appear

[10]G. R. Wood (1991), Multidimensional bisection and global optimization, *Computers and Math. Applic.* **21**, 161–172