Estimating Univariate Black Box Functions with Upper and Lower Bounds

Zelda B. Zabinsky
Industrial Engineering, Box 352650
University of Washington
Seattle, WA 98195

Robert L. Smith
Industrial and Operations Engineering
University of Michigan
Ann Arbor, Michigan 48109-2117

Birna P. Kristinsdottir
Mechanical and Industrial Engineering
University of Iceland
Hjardarhaga 2-6
107 Reykjavik, Iceland

Statement of scope and purpose

In engineering and science, it is often necessary to estimate functions based on a small number of evaluation points. We provide an estimation procedure that bounds a function using a Lipschitz bracket. We prove that the best sampling strategy is to sample at the midpoint of a specific interval.

Abstract

A procedure to estimate a univariate Lipschitz continuous function and provide upper and lower bounds is given in this paper. The procedure evaluates the function at the midpoint of the interval with the largest bracket. We provide a worst case analysis that proves this procedure is optimal in the sense that it maximizes the minimal removal region on each iteration, thus providing the tightest bounds in the least number of iterations possible.

Key words: Lipschitz functions, function estimation, interpolation, sampling strategy.

1 Introduction

Estimation of functions is an important topic in engineering. Interpolation (e.g. polynomial interpolation, splines, Bézier curves) has been used extensively [3, 4, 9] to create an estimating function based on a relatively small number of evaluation points. Interpolation is typically used when the function values are free of error. Interpolation or estimation is also useful when a function evaluation for a given point can be calculated, but the computation time may be significant.

In contrast to function evaluations that are free of error, function evaluations that contain errors due to random variation are often approximated by using regression analysis (e.g. least squares). Function evaluations containing error would include experimental data where different function evaluations are returned by the experiment at a given data value. In this paper, we restrict our consideration to the first category of functions that are free of error, sometimes referred to as black-box functions.

In estimating black-box functions, the large computation time motivates gaining as much information about the function with as few sample points as possible. The objective of this paper is to determine which points should be evaluated in order to maximize the information gained. Specifically, we prove a sampling algorithm that minimizes the maximum error.

Several authors have in particular looked at interpolation of Lipschitz continuous functions. In [8] a review of shape preserving approximation methods and algorithms is given. The primary focus is on interpolation methods by polynomials and splines. In [1] shape-preserving approximations and interpolation of function by box spline surfaces is studied. Other methods where the Lipschitz constant is used for function interpolation are discussed in [2] and [14]. In [2], the Lipschitz constant is used to get a linear approximation to a function and deriving a worst case error bound. In [14] the Lipschitz condition is used to create an octree representation that accelerates volume rendering.

The theory of Lipschitz continuous functions has also been used in global optimization, and is based on the fundamental work of [11] and [13]. More recent work involving Lipschitz methods for global optimization include [5, 6, 7, 10, 15, 16]. The emphasis of this paper is on estimation of the function, rather than optimization, and to provide tight bounds of the entire function based on only a few sample points.

In this paper the objective is to determine the best location of the sample points. The method used in this paper is to derive upper and lower bounds on a function $f(x)$ that is Lipschitz continuous. The data points and function evaluations, together with the Lipschitz constant can be used to bracket the function. As the upper and lower bounds get closer together, the area of the bracket is reduced, thus providing a better estimate of the function. In this paper, we obtain the location of sample points that minimizes the area of the bracket. This is equivalent to minimizing the maximum possible error for any function with that Lipschitz constant, as is explained in detail in the paper. We prove that the best estimate is obtained by sampling the midpoint of the interval with maximum volume. Although the Lipschitz constant is needed to provide the exact upper and lower bounds, the result that the optimal location of the sample point is at the midpoint of an interval is true, independent of the value of the Lipschitz constant. The benefits of estimating a function using the Lipschitz constant is that, in addition to the sample points, there exists an estimation function and a bracket that gives upper and lower bounds on the function. It is therefore guaranteed that the function can not have points outside these bounds.

2 Methodology

The function $f(x)$ that we want to estimate is a real valued function where $x \in I\!R$ and $f(x) : I\!R \to I\!R$. We assume that $f(x)$ satisfies the Lipschitz condition with a Lipschitz constant L, i.e.

$$|f(x) - f(y)| \leq L||x - y|| \text{ for all } x, y.$$

The Lipchitz condition assumes a bound on the rate of change of a function. When such a bound for a function exists, it is said to be Lipschitz continuous.

Given a Lipschitz continuous function $f(x)$ with a Lipschitz constant L over an interval $[a, b]$, we construct an estimating function $\hat{f}(x)$, and lower and upper bounding functions $\underline{F(x)}$ and $\overline{F(x)}$ respectively, such that $\underline{F(x)} \leq f(x) \leq \overline{F(x)}$ for any $x \in [a, b]$ with the following procedure:

Step 1 Initialize $x_1 = a$ and $x_2 = b$ and evaluate $f(x_1)$ and $f(x_2)$ to create an initial parallelogram. Set $k = 3$ and let $x_3 = (a + b)/2$ the midpoint of the interval. Evaluate $f(x_3)$.

Step 2 Update a list of parallelograms $P = \{v_1, v_2, \ldots, v_k\}$ where v_i is the leftmost coordinate for each parallelogram. Order the elements in the list by area, such that the first parallelogram has the maximum area. The area of a parallelogram with left x-coordinate v_i is calculated by:

$$\frac{(L^2(v_i - v_{i+1})^2 - (f(v_i) - f(v_{i+1}))^2)}{2L}. \tag{1}$$

Step 3 Check a termination criterion. if the first parallelogram has an area larger than δ, set $k = k+1$, evaluate the next iterate x_k at the midpoint of the first parallelogram (with maximum area) in the list P and go to Step 2. Otherwise, stop. The estimating function $\hat{f}(x)$ can be constructed from the sample points, for $i = 1, \ldots, k$:

$$\hat{f}(x) = \begin{cases} f(v_i) & \text{for } v_i \le x < v_i + d_i \\ Lx + \frac{L(v_i+v_{i+1})+f(v_i)+f(v_{i+1})}{2} & \text{for } v_i + d_i \le x < v_{i+1} - d_i \\ f(v_{i+1}) & \text{for } v_{i+1} - d_i \le x \le v_{i+1} \end{cases} \tag{2}$$

where $d_i = \frac{L(v_{i+1}-v_i)+f(v_{i+1})-f(v_i)}{2}$. The upper and lower bounds for $f(x)$ are:

$$\overline{F(x)} = \min_{i=1,2,\ldots,k} f(x_i) + L|x - x_i| \tag{3}$$

$$\underline{F(x)} = \max_{i=1,2,\ldots,k} f(x_i) - L|x - x_i| \tag{4}$$

Figure 1 shows $\hat{f}(x)$ and the upper and lower bounding functions after two iterations with function evaluations at x_1, x_2, x_3 and x_4. The bounding functions form three dark gray parallelograms and guarantee that the function will lie within this shaded bracket. The estimation function $\hat{f}(x)$ always lies equidistant between the bounding functions.

In step 2 a list of parallelograms is maintained, ordered by area. To illustrate using Figure 1, after the point x_3 is sampled we have two parallelograms. The left parallelogram with left coordinate x_1 has a larger area than the right parallelogram (with left coordinate x_3) because the difference in function values, $f(x_1) - f(x_3)$, is smaller than $f(x_3) - f(x_2)$:

$$\frac{L^2(\frac{a-b}{2})^2 - (f(x_1) - f(x_3))^2}{2L} > \frac{L^2(\frac{a-b}{2})^2 - (f(x_3) - f(x_2))^2}{2L}. \tag{5}$$

Figure 1: Successive bracketing of a function $f(x)$ using the Lipschitz constant.

Notice in this case the ordering of the parallelograms is determined solely by the function evaluations because the length of both intervals is the same ($\frac{a-b}{2}$). It is also interesting that the smaller the difference in function values, the larger the area of the parallelogram. In the extreme, if a function evaluation happened to land on the bounding function (the largest possible difference in function values), then the area of the parallelogram would be zero because the function must travel along the bounding function. This provides an intuitive explanation as to why it is less desirable to sample in that interval. Following the procedure with regard to Figure 1, the next point x_4 to be evaluated would be in the middle of the left interval, $x_4 = \frac{x_1+x_3}{2}$. The corresponding parallelograms would be added to the list. As more iterations are performed, the bracket tightens and provides a better estimate of the function. When the largest parallelogram is small enough, we terminate the method.

When selecting the next iteration point x_{k+1} it is not obvious which point is going to give the best approximation in the fastest way. Work done in global optimization [7] indicates that selecting the deepest point of some interval, but not necessarily the interval with the lowest bound, will give the fasted convergence of the optimization algorithms. This analysis uses an "average case" which is based on the assumption that possible points are uniformly distributed in the bracket. In the following section we use a worst case analysis to prove that, in fact, it is best to choose the midpoint (not necessarily the deepest point) of the interval with the largest parallelogram as the next iterate to remove the most area and hence tighten the bound and speed the approximation of the bracket.

3 Midpoint sampling

Performing successive bracketing of a function $f(x)$ results in a series of parallelograms as illustrated in Figure 1 and previously discussed. We now prove that the tightest bracket is obtained by sampling the midpoint of the interval with the largest parallelogram.

The objective of minimizing the area of the bracket is equivalent to minimizing the maximum possible estimation error. The error at any given point is the difference between the function estimate and the actual function value. The error could be as great as the difference between the values at the bracket, or function bounds. We choose our estimate as the midway point between the function bounds, which yields the estimation function given in equation 2. This function is flat until the first point of the parallelogram, which is equidistant between the upper and lower bounds. Then the function has a slope L until the peak, and then it is flat again. The estimation function is always half the distance between the Lipschitz bounds, and thus minimizes the maximum possible error. An estimate of the total possible error is one-half of the area of the parallelogram. As we continue to sample we tighten the Lipschitz bounds, generating more parallelograms with less total area than the original parallelogram. As we reduce the total area of the parallelograms, we reduce the worst case error of our estimate.

The proof is based on a worst case analysis. We first consider a single interval $[x_1, x_2]$ and characterize the area of the region that will be removed from the parallelogram if the function evaluated at point a in the interval is $f(a)$. This area is stated in Theorem 1. The next step in the analysis is to find the optimal location of point a in the interval. For any point a, there is a range of possible values for $f(a)$ in the bracket, and we want to perform a worst case analysis, i.e. maximize the minimal removal region. Theorem 2 states that the midpoint of the interval is the optimal place to sample. During the estimation procedure, a collection of intervals and corresponding parallelograms are generated. The final step in the analysis is to determine which interval should be sampled to provide the best estimate. It is shown in Theorem 3 and Corollary 1 that the interval with the largest parallelogram should be chosen to maximize the minimum removal region. Hence the procedure as stated in section 2 selects the best sample points to evaluate to obtain the tightest bracket on the function to be estimated.

Consider a parallelogram as illustrated in Figure 2. This parallelogram is formed by the upper and lower bounding functions generated by the function evaluated at x_1 and x_2. These create two pairs of parallel lines with absolute value slope of L and length l and h. We will consider the effect of evaluating the function at point a, where $x_1 \leq a \leq x_2$. The removal regionis defined as the regions above and below the point $(a, f(a))$ bounded by lines parallel to h and l as shown in figure 2. We can remove this region because by the Lipschitz condition, we know that f cannot lie in this region. We also let x_d denote the deepest point in the lower bounding function.

Let the angle whose tangent is $1/L$ be α as shown in Figure 2. We will also make use of the following relationships:

Figure 2: Removal region.

$$\sin\alpha = \frac{1}{\sqrt{1+L^2}} \tag{6}$$

$$\cos\alpha = \frac{L}{\sqrt{1+L^2}} \tag{7}$$

$$\sin 2\alpha = 2\cos\alpha\sin\alpha = \frac{2L}{1+L^2}. \tag{8}$$

We now state in Theorem 1 an expression for the area of the removal region, based on $(x_1, f(x_1))$, $(x_2, f(x_2))$ and the sample point $(a, f(a))$.

Theorem 1 *Given points $(x_1, f(x_1))$, $(x_2, f(x_2))$ and $(a, f(a))$, the area of the Total Removal Region, $TRR(a, f(a))$ is*

$$TRR(a, f(a)) = \frac{L^2(a-x_1)(x_2-a)-(f(a)-f(x_1))(f(x_2)-f(a))}{L} \tag{9}$$

where, x_1, x_2, a and $f(x_1)$, $f(x_2)$ and $f(a)$ are shown in Figure 2.

Proof.

To represent the area of the total removal region, we use the lower envelope function $\underline{F(x)} = f(x_i) - |x - x_i|$ to express two lines; $y_h = f(x_1)+L(x_1-x)$ and $y_l = f(x_2)+L(x-x_2)$ as illustrated in Figure 2. Similarly, the line y_m that is parallel to y_l and goes through the point $(a, f(a))$ can be expressed as $y_m = f(a) + L(x - a)$.

We determine x_i by finding where y_h and y_m intersect, yielding:

$$x_i = \frac{L(x_1 + a) + f(x_1) - f(a)}{2L}.$$

Similarily x_d the deepest point, is determined by finding where y_h and y_l intersect;

$$x_d = \frac{L(x_1 + x_2) + f(x_1) - f(x_2)}{2L}.$$

From x_i and x_d, we can also express the distance from x_1 to the first peak, which equals the distance from the deepest point to x_2, $d = x_i - x_1 = x_2 - x_d$. This is used in Step 3 of the algorithm.

We can now derive the lengths k, m, l and h (see Figure 2). Using $k\sin\alpha = x_d - x_i = x_j - a$ and equation 7 we have $k = (x_d - x_i)\sqrt{1+L^2}$, which gives

$$k = \frac{(L(x_2 - a) - f(x_2) + f(a))\sqrt{1+L^2}}{2L}.$$

Similarily $m \sin \alpha = x_j - x_d = a - x_i$ and therefore $m = (a - x_i)\sqrt{1+L^2}$, which gives

$$m = \frac{(L(a-x_1) + f(a) - f(x_1))\sqrt{1+L^2}}{2L}.$$

Similarily $l \sin \alpha = x_2 - x_d$ and therefore $l = (x_2 - x_d)\sqrt{1+L^2}$, which gives

$$l = \frac{(L(x_2-x_1) + f(x_2) - f(x_1))\sqrt{1+L^2}}{2L}.$$

Similarily $h \sin \alpha = x_d - x_1$ and therefore $h = (x_d - x_1)\sqrt{1+L^2}$, which gives

$$h = \frac{(L(x_2-x_1) + f(x_1) - f(x_2))\sqrt{1+L^2}}{2L}.$$

This gives

$$h - k = \frac{(L(a-x_1) - f(a) + f(x_1))\sqrt{1+L^2}}{2L}$$

$$l - m = \frac{(L(x_2-a) + f(x_2) - f(a))\sqrt{1+L^2}}{2L}.$$

The area of the total removal region is $TRR = km \sin 2\alpha + (h - k)(l - m) \sin 2\alpha = (km + (h-k)(l-m))(\frac{2L}{1+L^2})$, see figure 2.

Substituting for k, m, h, l and simplifying gives:

$$TRR(a, f(a)) = \frac{L^2(a-x_1)(x_2-a) - (f(a) - f(x_1))(f(x_2) - f(a))}{L}.$$

■

Using the expression for the total removal region given in Theorem 1, we next show that the midpoint is the best place to sample between x_1 and x_2 to maximize the minimal removal region.

Theorem 2 *The best place to sample in the interval* $[x_1, x_2]$ *in order to maximize the minimal area of the total removal region, is* $a^* = \frac{x_1+x_2}{2}$, *the mid point.*

Proof. For a given sample point a the function $f(a)$ can take any value on the line $x = a$ within the parallelogram. We first find $f(a)^*$ that gives the minimum total removal region for a fixed a, and then we find a^* that maximizes that minimal area (max worst case). For a fixed a, we minimize the removal region

$$TRR(a, f(a)) \quad = \quad \frac{L^2(a - x_1)(x_2 - a) - (f(a) - f(x_1))(f(x_2) - f(a))}{L}$$

by taking the partial derivative with respect to $f(a)$ and simplifying,

$$\frac{\partial TRR(a, f(a))}{\partial f(a)} = \frac{1}{L}(2f(a) - f(x_1) - f(x_2)).$$

Setting the above equation to zero and solving for $f(a)$ gives:

$$f(a)^* = \frac{f(x_1) + f(x_2)}{2}.$$

To check if this is the minimum, we take the second partial derivative,

$$\frac{\partial^2 TRR(a, f(a))}{\partial^2 f(a)} = 2,$$

which is positive. Therefore $TRR(f(a))$ is convex and $f(a)^* = \frac{f(x_1)+f(x_2)}{2}$ gives the minimum removal region as long as $\frac{f(x_1)+f(x_2)}{2}$ is in the interval defined by the upper and lower envelopes for $x = a$.

When $\frac{f(x_1)+f(x_2)}{2}$ is not within the parallelogram then $f(a)^*$ must be at either one of the bounding functions over which $f(a)$ can range within the parallelogram for a given a.

As illustrated in Figure 3, the bold line indicates the worst possible value $f(a)^*$ for a in the interval $[x_1, x_2]$ that minimizes the removal region. Given this worst case $f(a)^*$, we now find the value for a that maximizes the worst case removal region. We again consider two cases; the first case is when $f(a)^* = \frac{f(x_1)+f(x_2)}{2}$ is inside the parallelogram for a given a, the second case is when $f(a)^*$ is on the envelope.

Consider the first case where $f(a)^* = \frac{f(x_1)+f(x_2)}{2}$ is inside the parallelogram. The minimum removal region for a given a at $f(a)^*$ is

$$TRR(a, f(a)^*) = \frac{4L^2(a - x_1)(x_2 - a) - (f(x_2) - f(x_1))^2}{4L}. \qquad (10)$$

Figure 3: For $a \in [x_1, x_2]$, the thick line indicates the value of $f(a)^*$ that is the worst case, i.e. has minimal removal region.

Taking the partial derivative with respect to a gives

$$\frac{\partial TRR(a, f(a)^*)}{\partial a} = L(-2a + x_1 + x_2)$$

and setting equal to zero and solving for a gives

$$a^* = \frac{x_1 + x_2}{2}.$$

This is the midpoint of the interval $[x_1, x_2]$.

In the second case when $f(a)^*$ is on the envelope, it is clear that the removal region for a in these intervals is smaller than the removal region at the midpoint. In fact, the removal region decreases as a^* is further from the midpoint. Hence it is clear that the midpoint will give the maximum removal region. ■

The following theorem expresses the area of a parallelogram, and Corollary 1 states that the best parallelogram to sample is the one with the largest area.

Theorem 3 *The area of a parallelogram defined over the interval $[x_1, x_2]$ with function values $f(x_1)$ and $f(x_2)$ can be expressed as*

$$\frac{(L^2(x_1 - x_2)^2 - (f(x_1) - f(x_2))^2)}{2L}.$$

Proof. Consider a parallelogram over the interval $[x_1, x_2]$, where the leftmost corner has the coordinates $(x_1, f(x_1))$ and the rightmost corner has the coordinates $(x_2, f(x_2))$. The area of the parallelogram (see Figure 4) can be expressed as

$$hl \sin 2\alpha = \frac{2Lhl}{1 + L^2}.$$

Using several identities given in equations (3), (4) and (8) we can express h and l in terms of the coordinates $(x_1, f(x_1))$, $(x_2, f(x_2))$ and the Lipschitz constant.

The lower bracketing function that goes through points $(x_1, f(x_1))$ and $(x_d, f(x_d))$ has the representation $y_h = f(x_1) - L|x - x_1| = L(x_1 - x) + f(x_1)$ for $x \in [x_1, x_d]$. Similarly the lower bracketing function that goes through points $(x_d, f(x_d))$ and $(x_2, f(x_2))$ can be written as $y_l = f(x_2) - L|x - x_2| = L(x - x_2) + f(x_2)$ for $x \in [x_d, x_2]$.

The two lines intersect at x_d, and we find the intersection point by letting $y_h = y_l$ and solving for x, which gives:

$$x_d = \frac{L(x_1 + x_2) + f(x_1) - f(x_2)}{2L}.$$

Using that $h = (x_d - x_1)\sqrt{1 + L^2}$ and using the expression for x_d just derived gives

$$h = \frac{(-L(x_1 - x_2) + (f(x_1) - f(x_2)))\sqrt{1 + L^2}}{2L}.$$

Similarily, for l we get $l = (x_2 - x_d)\sqrt{1 + L^2}$, or

$$l = \frac{(-L(x_1 - x_2) - (f(x_1) - f(x_2)))\sqrt{1 + L^2}}{2L}$$

after simplifying, the area of the parallelogram can be expressed as

$$\frac{2Lhl}{1 + L^2} = \frac{(L^2(x_1 - x_2)^2 - (f(x_1) - f(x_2))^2)}{2L}.$$

■

Corollary 1 *Given a sequence of points $v_1, v_2, \ldots, v_{n+1}$ and n parallelograms, the next parallelogram to use for sampling to maximize the minimal total removal region is the parallelogram with the largest area.*

Proof. For a given interval $[x_1, x_2]$, the midpoint, $a^* = \frac{x_1 + x_2}{2}$ maximizes the minimal removal region and gives the worst case removal region as

$$TRR(a^*, f(a)^*) = \frac{L^2(x_2 - x_1)^2 - (f(x_2) - f(x_1))^2}{4L}.$$

The area of any parallelogram is $\frac{L^2(x_2 - x_1)^2 - (f(x_2) - f(x_1))^2}{2L}$, and therefore the total removal region is maximized by picking the interval with the largest parallelogram. ■

Figure 4: Calculating the area of a parallelogram.

4 Conclusions

The procedure presented in section 2 provides an estimate and bounds for a Lipschitz continuous function by sampling the midpoint of the interval with the largest bracket. Section 3 presents a worst case analysis that proves this procedure is optimal in the sense that, on each iteration, the choice of point to be evaluated maximizes the minimal removal region. This is equivalent to minimizing the maximum possible error of the estimate. This single procedure provides an efficient and effective means to provide a tight bracket to estimate a Lipschitz function.

5 Acknowledgements

This work has been funded in part by an NSF collaborative grant, numbers DMI-9820878 and DMI-9820744. The authors want to acknowledge Ivone Sasmitra for her early work during her masters thesis [12]. We also thank Jana Littleton and Jennifer Temple for their assistance during the revision of this paper.

References

[1] C.K. Chui, H. Diamond, L. Raphael. Shape-preserving quasi-interpolation and interpolation by box spline surfaces. Journal of Computational and Applied Mathematics 1989; 25: 169-198.

[2] D.A. Cooper. Learning Lipschitz Functions. International Journal of Computer Mathematics 1995; 59: 15-26.

[3] P. Dierckx. Curve and Surface Fitting with Splines. Oxford Science Publications. New York. 1993.

[4] P. Lancaster, K. Salkauskas. Curve and Surface Fitting: An Introduction. Academic Press. London. 1986.

[5] P. Hansen, B. Jaumard, S-H. Lu. Local optimization of univariate Lipschitz functions: I. Survey and properties. Mathematical Programming 1992; 55: 251-272.

[6] P. Hansen, B. Jaumard, S-H. Lu. Global optimization of univariate Lipschitz functions: II. New algorithms and computational comparison. Mathematical Programming 1992; 55: 273-292.

[7] P. Hansen, B. Jaumard, S-H. Lu. On Timonov's Algorithm for Global Optimization of Univariate Lipschitz Functions. Journal of Global Optimization 1991; 1: 47-64.

[8] L.M. Kocic, G.V. Milovanovic. Shape-preserving Approximations by Polynomials and Splines. Computers and Mathematics with Applications, 1997; 33(11): 59-97.

[9] D. Kahaner, C. Moler, S. Nash. Numerical Methods and Software. Prentice Hall Series in Computational Mathematics. Prentice-Hall, 1989.

[10] J.D. Pinter. Global Optimization in Action, Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications. Kluwer Academic Publishers, 1996.

[11] S.A. Piyavskii. An Algorithm for Finding the Absolute Extreme of a Function. USSR Computational Mathematics and Mathematical Physics 1972; 12: 57-67.

[12] I. Sasmitra. Empirical and Analytical Comparisons of Deepest Point with Stochastic Lipschitz Optimization. Masters Thesis, University of Washington, 1995.

[13] B.O. Shubert. A Sequential Method Seeking the Global Maximum of a Function. SIAM Journal on Numerical Analysis, 1972; 9: 379-388.

[14] B.T. Stander, J.C. Hart. A Lipschitz Method for Accelerated Volume Rendering. Proceedings of 1994 Symposium on Volume Visualization. IEEE, New York, NY, USA. 1995. p 107-114.

[15] R.J. Vanderbei. Extension of Piyavskii's Algorithm to Continuous Global Optimization. Journal of Global Optimization, 1999; 14: 205-216.

[16] G.R. Wood. The Bisection Method in Higher Dimensions. Mathematical Programming. 1992; 55: 319-337.