Computing 28, 225—237 (1982)

Computing

On Computing the Range of Values

N. S. Asaithambi, Shen Zuhe, and **R. E. Moore,** Madison, Wisconsin

Received April 7, 1981; revised September 28, 1981

Abstract — Zusammenfassung

On Computing the Range of Values. A simple algorithm is given for computing the range of values of a differentiable function over an n-dimensional rectangle.

Die Berechnung des Wertebereichs. Gegeben ist ein einfaches Verfahren zur Berechnung des Wertebereichs einer differenzierbaren Funktion auf einem n-dimensionalen Quader.

1. Introduction

A problem of fundamental importance in computing is that of finding good upper and lower bounds on the range of values of a function of several variables in an n-dimensional rectangle. This includes: finding bounds on the solution to a problem with initial data or contants known only to lie in certain intervals (sensitivity analysis, perturbation analysis), finding the ranges of remainder terms and bounds on norms of functions and operators in error analysis, and finding minimum or maximum values in mathematical programming problems.

Techniques of *interval analysis* have been developed for just such purposes (Moore (1976, 1979)).

In Moore (1979, p. 49) a simple algorithm using "cyclic bisection" was suggested. Unfortunately, however, as pointed out by N. S. Asaithambi, the algorithm presented there contains errors. In this paper, we give an algorithm which avoids those errors and is almost equally simple. In particular, our algorithm does not use the matrix of second partial derivatives and so is simpler than the method proposed by Hansen (1980). On the other hand, the method of Hansen is designed to find the *location* of the global minimum and, while our method can be adapted to do that also, the method of Hansen is undoubtedly more efficient for that purpose.

We consider the problem of computing upper and lower bounds on the range of values

0010-485X/82/0028/0225/$ 02.60

$$f(X^{(0)}) = \{f(x_1, x_2, \ldots, x_n) : x_i \in x_i^{(0)},\ i = 1\,(1)\,n\} = [\underline{f(X^{(0)})}, \overline{f(X^{(0)})}], \tag{1.1}$$

where

$$X^{(0)} = (X_1^{(0)}, \ldots, X_n^{(0)})$$

is an n-dimensional vector of intervals

$$X_i^{(0)} = [\underline{X_i^{(0)}}, \overline{X_i^{(0)}}],\ i = 1\,(1)\,n,$$

and f is continuous in $X^{(0)}$.

We suppose that we have an interval extension (Moore (1979)) F of the function f. Thus, if $x \in X \subseteq X^{(0)}$, then $f(x) \in F(X)$. We can compute $F(X^{(0)})$ and obtain, with a single evaluation of F, an interval containing the range of values of f:

$$f(X^{(0)}) \subseteq F(X^{(0)}) = [\underline{F(X^{(0)})}, \overline{F(X^{(0)})}]. \tag{1.2}$$

However, the width of $F(X^{(0)})$ may exceed the width of the exact range of values (1.1) by an unacceptable amount.

We define the *excess width* of an interval bound $[a, b]$ on the ranges of values $f(X^{(0)}) \subseteq [a, b]$, as:

$$E = b - \overline{f(X^{(0)})} + \underline{f(X^{(0)})} - a = w([a, b]) - w(f(X^{(0)})). \tag{1.3}$$

Thus, if $f(X^{(0)}) \subseteq [a, b]$, then the excess width of $[a, b]$ is the sum of the differences between corresponding endpoints. As a result of the definitions (1.1) and (1.3), a and b are respectively the minimum and maximum values of f in $X^{(0)}$ if and only if $E = 0$.

What we seek is a simple, reasonably efficient algorithm for finding an interval containing $f(X^{(0)})$ with *arbitrarily small* excess width – in other words, an algorithm which converges to the exact range of values.

If F has the reasonable properties

1. $X \subseteq Y \subseteq X^{(0)}$ implies $F(X) \subseteq F(Y)$ [F is inclusion monotonic],
2. for some L and all $X \subseteq X^{(0)}$, $w(F(X)) \leq L\, w(X)$ [F is Lipschitz], where $w(X)$ is the width of the interval X,

then we can subdivide each $X_i^{(0)}$ into N sub-intervals of equal width and obtain (Moore (1979)):

$$\bigcup_j F(X^{(j)}) = f(X^{(0)}) + E \tag{1.4}$$

with $w(E) \leq L\, w(X^{(0)})/N$. The union is taken over all N^n parts of the subdivision.

While this result enables us to make the excess width E arbitrarily small by taking N large enough, it would require an enormous amount of computation for large values of N even if n is fairly small.

We can make a vast reduction in the amount of computation required by using an idea which is due to Skelboe (1974). In this paper, we further reduce the amount of computation required by modifying the method of Skelboe in a number of ways.

Suppose we subdivide the initial region $X^{(0)}$ *into two* parts by bisecting it in *one* of the coordinate directions so that $X^{(0)}$ can be written as the union of two regions $X^{(1)}$ and $X^{(2)}$. If we find that $F(X^{(1)}) < v$ where v is the value of $f(x)$ for some x in $X^{(2)}$, say the midpoint of $X^{(2)}$, then we can *exclude* the entire region $X^{(1)}$ from further consideration in seeking an upper bound on the range of values of f. The maximum value of f *must* lie in $X^{(2)}$. We call this the *midpoint test*. However, if v is contained in $F(X^{(1)})$, then further subdivisions of both $X^{(1)}$ and $X^{(2)}$ may be necessary.

In the next section, we present an algorithm which makes use of this idea and others to compute the range of values in a reasonably efficient manner.

The cyclic bisection method proposed in Moore (1979, p. 49) has at least three flaws. First, as noticed by N. S. Asaithambi, step (9) should follow step (11) and should read: set $b_0 = \underline{F}(X)$. But even with this change, there are two additional problems. Shen Zuhe has constructed simple two-dimensional examples for which the cyclic choice of coordinate directions in which bisections are to be done produces a sequence of thinner and thinner slit-shaped regions so that the method does not appear to converge. The third problem we have found is that the method may stop prematurely (while still far from the exact range of values). By bisecting a region in a coordinate direction in which the region has maximum width, the second problem disappears. The third problem was the most troublesome. In order to avoid premature stopping, we must be sure we have bisected (possibly different regions) in each of a certain set of coordinate directions at least once. There may be one or more coordinate directions in which bisection does not change the lower bound at all; whereas bisection in other directions may still be increase the lower bound.

2. Problem with Premature Convergence

The third problem with cyclic bisection is the danger of premature convergence. The problem was detected when we attempted to use the cyclic bisection method to find the range of values of

$$f(x_1, x_2, x_3) = \frac{x_1 + x_2}{x_1 - x_2} x_3 \tag{2.1}$$

where $x_1 \in X_1 = [1, 2]$, $x_2 \in X_2 = [5, 10]$ and $x_3 \in X_3 = [2, 3]$.

The actual range of (2.1) can be easily computed as $[-7, \frac{22}{9}]$. With the natural interval extension, the cyclic bisection method converged with a single bisection yielding a lower bound of -12 instead of -7.

A straight forward substitution for x_1, x_2, x_3 in (2.1) with X_1, X_2, X_3 as intervals yields a lower bound of -12. This means that the condition $b \le b_0$ is satisfied at the very beginning and hence the procedure stops prematurely. It is found that

$$b_0 = \underline{F}(X_0) = -12;$$

$$\underline{F}(X_0^{(1)}) = -9.857, \ \underline{F}(X_0^{(2)}) = -12;$$

$$b = \min(-9.857, -12) = -12 = b_0.$$

$(X_0^{(1)}, X_0^{(2)}$ resulting after a bisection along direction 1), hence the first $b=b_0$ is no indication that we have hit the lower bound and the condition of step (7) should be altered. On the other hand, for a *rational* function, the first $b=b_0$ sometimes *does* indicate that we have hit the lower bound. The following results are useful in this respect.

Let f be said to be in *rational canonical form* if it can be expressed as

$$f(x)=\frac{a_0+a_1x+\ldots+a_nx^n}{b_0+b_1x+\ldots+b_mx^m}$$

for a one dimensional case and a similar form for n-variables.

Lemma 2.1: *Suppose X is an interval, $p(x)$ is a polynomial canonical form on X. $P(X)$ is its natural extension. Let $b_0=\underline{P(X)}$. Suppose $X=X^{(1)}\cup X^{(2)}$. If $b=\mathrm{Min}\left(\underline{P(X^{(1)})}, \underline{P(X^{(2)})}\right)=b_0$, then there exists an $x\in X$ such that*

$$p(x)=b_0.$$

Proof: Let

$$p(x)=\sum_{i=0}^{n}\alpha_i x^i.$$

For the abbreviation of notation, set

$I=\{i\,|\,\alpha_i\geq 0,\ i=0,1,\ldots,n\}$;

$II=\{i\,|\,\alpha_i<0,\ i=0,1,\ldots,n\}$;

$III=\{i\,|\,i=\text{odd},\ i=0,1,\ldots,n\}$;

$IV=\{i\,|\,i=\text{even},\ i=0,1,\ldots,n\}$;

First of all, suppose $X\geq 0$, $X=[\underline{X},\bar{X}]$, then

$$\underline{P(X)}=\sum_I \alpha_i[\underline{X},\bar{X}]^i+\sum_{II}\alpha_i[\underline{X},\bar{X}]^i$$

and

$$b_0=\underline{P(X)}=\sum_I \alpha_i\underline{X}^i+\sum_{II}\alpha_i\bar{X}^i. \tag{2.2}$$

If $b_0=\underline{P(X^{(1)})}$, then

$$b_0=\sum_I \alpha_i\underline{X}^i+\sum_{II}\alpha_i m(X)^i. \tag{2.3}$$

From (2.2) and (2.3), we have

$$\sum_{II}\alpha_i(\bar{X}^i-m(X)^i)=0.$$

We must have $\alpha_i=0$ $(i\in II)$, since $\bar{X}-m(X)^i>0$, $(i\in II)$; therefore

$$p(\underline{X})=b_0.$$

If $b_0=\underline{P(X^{(2)})}$, then

$$b_0=\sum_I \alpha_i m(X)^i+\sum_{II}\alpha_i\bar{X}^i. \tag{2.4}$$

From (2.2) and (2.4), we have

$$\sum_{I} \alpha_i (\underline{X}^i - m(X)^i) = 0,$$

therefore

$$\alpha_i = 0\,(i \in I)$$

and

$$b_0 = p(\bar{X}).$$

For X≤0, the proof is the same as above.

Now suppose $X = [\underline{X}, \bar{X}]$, $\underline{X} < 0 < \bar{X}$, then

$$P(X) = \sum_{I\,III} \alpha_i [\underline{X}^i, \bar{X}^i] + \sum_{I\,IV} \alpha_i [0, |X|^i] + \sum_{II\,III} \alpha_i [\underline{X}^i, \bar{X}^i] + \sum_{II\,IV} \alpha_i [0, |X|^i]$$

and

$$b_0 = \underline{P(X)} = \sum_{I\,III} \alpha_i \underline{X}^i + \sum_{II\,III} \alpha_i \underline{X}^i + \sum_{II\,IV} \alpha_i |X|^i. \tag{2.5}$$

If $m(X) < 0$ and $b_0 = \underline{P(X^{(1)})}$, then

$$P(X^{(1)}) = \sum_{I\,III} \alpha_i [\underline{X}^i, m(X)^i] + \sum_{I\,IV} \alpha_i [m(X)^i, \underline{X}^i] + \sum_{II\,III} \alpha_i [\underline{X}^i, m(X)^i] + \sum_{II\,IV} \alpha_i [m(X)^i, \underline{X}^i]$$

and

$$b_0 = \underline{P(X^{(1)})} = \sum_{I\,III} \alpha_i \underline{X}^i + \sum_{I\,IV} \alpha_i m(X)^i + \sum_{II\,III} \alpha_i m(X)^i + \sum_{II\,IV} \alpha_i \underline{X}^i. \tag{2.6}$$

From (2.5) and (2.6), we have

$$-\sum_{I\,IV} \alpha_i m(X)^i + \sum_{II\,III} \alpha_i (\bar{X}^i - m(X)^i) + \sum_{II\,IV} \alpha_i (|X|^i - \underline{X}^i) = 0,$$

therefore

$$\alpha_i = 0, \; (i \in I\;IV,\; II\;III,\; II\;IV),$$

and

$$b_0 = p(\underline{X}).$$

The proof for the cases $b_0 = \underline{P(X^{(2)})}$ and $m(X) > 0$ is exactly the same.

Note: Lemma 2.1 is no longer valid in the case $m(X) = 0$. However, we can change the subdivision of $X = X^{(1)} \cup X^{(2)}$ and avoid this case.

Lemma 2.2: *In addition to polynomial $p(x)$, lemma 2.1 is valid for any function in rational canonical form on X.*

Proof: Suppose

$$r(x) = \frac{p(x)}{q(x)}.$$

Without loss of generality, suppose $q(x)>0$. Because of the similarity of other cases, we only give the proof of the case $X>0$ and $P(X)>0$. In this case

$$R(X)=\frac{[\underline{P(X)},\overline{P(X)}]}{[\underline{Q(X)},\overline{Q(X)}]}$$

and

$$b_0=\underline{R(X)}=\frac{\underline{P(X)}}{\overline{Q(X)}}.$$

Since $b_0=\min(\underline{R(X^{(1)})},\underline{R(X^{(2)})})$, if $b_0=\underline{R(X^{(1)})}$, then

$$\frac{\underline{P(X)}}{\overline{Q(X)}}=\frac{\underline{P(X^{(1)})}}{\overline{Q(X^{(1)})}}.$$

Because

$$\underline{P(X)}\le\underline{P(X^{(1)})},\ \overline{Q(X)}\ge\overline{Q(X^{(1)})},$$

so

$$\underline{P(X)}=\underline{P(X^{(1)})},\ \overline{Q(X^{(1)})}=\overline{Q(X)},$$

and

$$r(\underline{X})=b_0.$$

If $b_0=\underline{R(X^{(2)})}$, then $r(\overline{X})=b_0$. Lemma 2.2 is proved.

Lemma 2.3: *In addition to rational functions $r(X)$, lemma 2.2 is valid for any functions in rational canonical form with interval coefficients.*

The above results are also valid in the case of n-variables; so we have

Theorem 2.1: *Suppose that*

$$X=(X_1,X_2,\ldots,X_n)\subseteq R^n,\ x_i\in X_i;\ f(x_1,x_2,\ldots,x_n)$$

is a rational function on X;

$$F(X_1,X_2,\ldots,X_n)$$

is its natural extension;

$$b_0=\underline{F(X_1,X_2,\ldots,X_n)}.$$

If for some i $(1\le i\le n)$, $X_i=X_i^{(1)}\cup X_i^{(2)}$, with

$$b_0=\min\big(\underline{F(X_1,X_2,\ldots,X_{i-1},X_i^{(1)},X_{i+1},\ldots,X_n)},$$

$$\underline{F(X_1,X_2,\ldots,X_{i-1},X_i^{(2)},X_{i+1},\ldots,X_n)}\big);$$

then there exists an $x_i\in X_i$ such that

$$b_0=\underline{F(X_1,\ldots,X_{i-1},x_i,X_{i+1},\ldots,X_n)}.$$

Corollary: *Suppose*

$$b_0 = \underline{F(X_1, X_2, \ldots, X_n)}, \; X_i = X_i^{(1)} \cup X_i^{(2)}.$$

If for $i = 1\,(1)\,n$,

$$b_0 = \min\left(\underline{F(x_1^0, \ldots, x_{i-1}^0, X_i^{(1)}, X_{i+1}, \ldots, X_n)},\right.$$

$$\left.\underline{F(x_1^0, \ldots, x_{i-1}^0, X_i^{(2)}, X_{i+1}, \ldots, X_n)}\right),$$

then

$$f(x_1^0, \ldots, x_n^0) = \min_{x \in X} f(x_1, \ldots, x_n).$$

Since "$b = b_0$" with the natural interval extension F of f, when f is a function in rational canonical form, ensures that one of the arguments is reduced to a point from an interval; repeatedly applying these results for each subsequent reduced problem, one can either end up with a single point, in which case, the lower or upper bound is readily known, or a much smaller region over which bisection can be carried out to get the bound. Therefore, this theorem can be used to reduce the dimensionality of the problem or avoid the premature convergence. Putting all these things together, we conclude that, before starting any search for the bound, it would better to use this theorem to reduce (if it is possible) the region X to $\hat{X}$; if $\hat{X}$ is not a single point, in general, an m-dimensional region ($m < n$) is produced for further search to obtain the bound.

3. Finding the Range of Values

The ideas behind the algorithm to be presented in this section are these:

1. we seek first a good *lower* bound on the *minimum* value of f over $X^{(0)}$; we then apply the same method to $-f$ in order to find a good *upper* bound on the *maximum* value of f over $X^{(0)}$; this follows the strategy suggested by Skelboe (1974);

2. from the initial region $X^{(0)}$, we generate a *list* of subregions whose union *must* contain the global minimum; the elements in the list are generated in *pairs*; each pair is the result of a bisection in a *single* coordinate direction of some previous region on the list; the elements are entered into the list in order of increasing lower bounds so that the *first* element in the list always corresponds to the *lowest* current lower bound; it will be examined next (this also is Skelboe's strategy); this differs from Skelboe's method in that he bisected simultaneously in *all* coordinate directions to generate 2^n new regions at each step; we generate only two new regions;

3. we choose, as our direction in which to bisect a region X, the first direction in which X has maximum width; this prevents slit-shaped regions and insures that the diameters of the regions will decrease;

4. we do not list a region at all if it can be determined by the midpoint test (see section 1) that the region cannot contain the global minimum of f on $X^{(0)}$; we also

delete all elements on the list which would follow such an element if it were inserted in proper order;

5. we assume that f is continuously differentiable in $X^{(0)}$ and that we have interval extensions $D_i F$ of the partial derivatives $\partial f/\partial x_i$, $i=1(1)n$; we use monotonicity tests to *reduce the dimension* of a region as much as possible before listing it; thus, instead of listing X, we can list Y with: $Y_i = \underline{X}_i$ if $D_i F(X) \geq 0$ or $Y_i = \bar{X}_i$ if $D_i F(X) \leq 0$ or $Y_i = X_i$ otherwise, where $X_i = [\underline{X}_i, \bar{X}_i]$, $i=1(1)n$;

6. we make use of the *mean value form*

$$F_{MV}(X) = f(m) + \sum_{i=1}^{n} D_i F(X)(X_i - m_i), \tag{3.1}$$

where m is the midpoint of X, for an interval extension of f (Moore (1979)); Alefeld and Herzberger (1974, Satz 6) have shown that the excess width of $F_{MV}(X)$ is of the order $w(X)^2$; thus, for small regions X, as we come close to convergence to the minimum value of f, the convergence will be accelerated by use of F_{MV} instead of a "natural" extension of f; actually, we combine the mean value form with the monotonicity tests and use the interval extension (*monotonicity test form*, Moore (1979))

$$F_{MT}(X) = [f(u), f(v)] + \sum_{i \in S} D_i F(X)(X_i - m_i) \tag{3.2}$$

where S is the set of integers i such that $D_i F(X)$ properly contains zero and

$$(u_i, v_i) = \begin{cases} (\underline{X}_i, \bar{X}_i) & \text{if } \underline{D_i F(X)} \geq 0 \\ (\bar{X}_i, \underline{X}_i) & \text{if } \overline{D_i F(X)} \leq 0 \\ (m_i, m_i) & \text{if } i \in S; \end{cases}$$

7. we *terminate* the computation when there is no further increase in the current lower bound; *this will always happen in a finite number of steps* using rounded interval arithmetic, Moore (1979);

8. during the computation of $F_{MT}(X)$ defined by (2.2), we also find the set S and the *reduced region* Z with components Z_i given by

$$Z_i = \begin{cases} [u_i, u_i] & \text{for } i \notin S \text{ with } u_i \text{ defined by (3.2)} \\ X_i & \text{for } i \in S. \end{cases} \tag{3.3}$$

Our algorithm for finding the range of values is as follows:

We first find a lower bound on the range of values of f over $X^{(0)}$ (*see* (1.1)); we then apply the same procedure to find a lower bound on $-f$ over $X^{(0)}$ to get the upper bound;

step 1: compute $b_0 = \underline{F_{MT}(X^{(0)})}$, S, and $Z^{(0)}$;

step 2: if S is empty, stop;

step 3: set $X = Z^{(0)}$;

step 4: initially, the *list* is empty;

step 5: let $w = \max_{i \in S} w(X_i)$; if $w = 0$, stop; otherwise set
i = the first element of S such that $w(X_i) = w$;

step 6: bisect X in direction i to obtain $X^{(1)}$ and $X^{(2)}$;

step 7: compute $b_1 = \underline{F_{MT}(X^{(1)})}$, $S^{(1)}$, and $Z^{(1)}$;

step 8: if $S^{(1)}$ is empty and $Z^{(1)}$ does not intersect the boundary of $Z^{(0)}$, discard $Z^{(1)}$ and to to *step 12*;

step 9: apply the midpoint test using region $Z^{(1)}$ to delete all unnecessary elements from the *list* (if any);

step 10: insert ($Z^{(1)}$, b_1, $S^{(1)}$) into the *list* in *proper order* (of increasing second components – thus lowest b is first);

step 11: if *list* overflows, print message and stop;

step 12: compute $b_2 = \underline{F_{MT}(X^{(2)})}$, $S^{(2)}$, and $Z^{(2)}$;

step 13: if $S^{(2)}$ is empty and $Z^{(2)}$ does not intersect the boundary of $Z^{(0)}$, discard $Z^{(2)}$ and go to *step 17*;

step 14: apply the midpoint test using region $Z^{(2)}$ to delete all unnecessary elements from the *list* (if any);

step 15: insert ($Z^{(2)}$, b_2, $S^{(2)}$) into the *list* in proper order;

step 16: if *list* overflows, print message and stop;

step 17: set $b = b_1$ if $Z^{(1)}$ was listed but not $Z^{(2)}$;
set $b = b_2$ if $Z^{(2)}$ was listed but not $Z^{(1)}$;
set $b = \min(b_1, b_2)$ if both $Z^{(1)}$ and $Z^{(2)}$ were listed;

step 18: if neither $Z^{(1)}$ nor $Z^{(2)}$ was listed, go to *step 20*;

step 19: if $b \leq b_0$, stop;

step 20: if the *list* is empty, stop;

step 21: set b_0 = lower bound (second component) in the first element on the list (see *steps 10* and *15*);

step 22: set X = region (first component) in the first element on the list; S = index set (third component) in the first element on the list;

step *23*: go to *step 5*.

Comments: If the computation stops at step 2, then f is monotonic in all directions and b_0 is the minimum value and f takes on its minimum value over $X^{(0)}$ at the point u defined in (3.2); this will, of course, be a boundary point. If we stop at step 5, then the global minimum value occurs at the *point* X. If the list overflows at step 11 or step 16, then we would need more storage space allocated for the list in order to do the problem. If the computation stops at step 19, then further bisections will not improve the lower bound using the finite precision rounded interval arithmetic employed; a sharper lower bound might be found by going to higher precision interval arithmetic. Finally, if the computation should stop at step 20, then we have found the lower bound on the range of values. In any case, when the computation stops (except for storage overflow at steps 11 or 16), the value b_0 is a lower bound on the range of values of f over $X^{(0)}$. When the procedure is applied to $-f$, the value b_0 is an upper bound to the range of values of f over $X^{(0)}$.

The computation *will always stop* after a *finite* number of steps. In fact, unless the computation stops at steps 2, 5, 11, 16, or 20, it *must* stop at step 19 after some *finite* number of executions of various steps because we obtain a monotonic[1] increasing sequence of lower bounds, bounded above by $\underline{f(X^{(0)})}$, contained in a finite set of machine numbers.

Thus, we have proved the following.

Theorem: *The algorithm given by steps* 1 – 23 *always converges in a finite number of computational steps to an interval containing the range of values of over* $X^{(0)}$.

How close to the exact range of values the resulting interval will be depends on the precision of the arithmetic we use and on the particular function f and the region $X^{(0)}$.

4. Numerical Results

In this section we will report on a number of examples we have run using the algorithm of the previous section. The computations have been carried out on the UNIVAC 1110 computer using single precision (about 8 decimals) interval arithmetic with directed rounding (see Moore (1979)).

For each example we will give: the *function f*, the *region* $X^{(0)}$, two integers indicating the maximum *list sizes* which occurred during the computation of the minimum and maximum values of f, the total number of *bisections* required, and the interval *bounds* on the range of values. The full algorithm and a simplified version without the midpoint tests (steps 9 and 14) gave almost exactly the same results for all but the last example we will show. Therefore, we only indicate the results for the simplified version for the final example.

Example 1: $f(x)=x(1-x)$

Region	List sizes	Bisections	Bounds
[0, 1]	2, 2	2	[0, 0.25]
[−0.5, 1]	2, 1	17	[−0.75, 0.25]
[−1, 1]	2, 2	3	[−2, 0.25]
[−2, 6]	2, 2	5	[−30, 0.25]
[−3, 6]	2, 1	20	[−30, 0.25]

[1] Until $b \leq b_0$ as tested in step 19.

Example 2: $f(x)=x_1(1-x_1)(1-5/8\,x_2+3/2\,x_2^2-x_2^3)$

Region	List sizes	Bisections	Bounds
([0, 1], (0, 1])	12, 16	26	[0, 0.25]
([0, 1], [−1, 1])	14, 6	18	[0, 1.03125]
([−1, 1], [0, 1])	11, 18	30	[−2, 0.25]
([−1, 1], [−1, 1])	4, 9	14	[−8.25, 1.03125]

Example 3: $f(x)=(1-x_1)(x_1-5/8\,x_1\,x_2+3/2\,x_1^2\,x_2^2-x_1^3\,x_2^3)$

Region	List sizes	Bisections	Bounds
([0, 1], [0, 1])	30, 68	118	[0, 0.25]
([0, 1], [−1, 1])	32, 37	75	[0, 0.69279174]
([−1, 1], [0, 1])	5, 4	7	[−2, 4.25]
([−1, 1], [−1, 1])	19, 8	28	[−2.25, 4.25]

Example 4: The so-called "three hump camel function"; see Hansen (1980):

$$f(x)=2\,x_1^2-1.06\,x_1^4+1/6\,x_1^6-x_1\,x_2+x_2^2,$$

which has three minima and two saddle points in

$$X^{(0)}=([-2,4],\ [-2,4]).$$

Region	List sizes	Bisections	Bounds
([−2, 4], [−2, 4])	65, 13	212	$[-1.8208\times 10^{-14}$, $457.86667]$

The simplified version (with midpoint tests) required 416 bisections, by comparison. The global minimum was found to lie in the two dimensional rectangle given by

$$([-2.9802322\times 10^{-8},\ 5.9604645\times 10^{-8}],$$
$$[-2.9802322\times 10^{-8},\ 5.9604645\times 10^{-8}]).$$

The exact range of values of the function in example 4 over the given $X^{(0)}$ is contained in and very close to the computed bounds shown above: $[-1.8208\times 10^{-14}, 457.86667]$. In contrast to this, if we were to subdivide $X^{(0)}$ in both the x_1 and x_2 direction and evaluate the natural interval entension for each of the N^2 parts according to (1.4), it would require some 10^{15} evaluations to reduce the excess width to about 10^{-5}. The algorithm given in this paper produces such a result in 424 evaluations of (3.2) – two for each bisection.

Example 5: $f(x_1,x_2,\ldots,x_n)=f_1(x_1)f_2(x_2)f_3(x_3)f_4(x_4)f_5(x_5)$ where

$$f_1(x_1)=0.01\,x_1(x_1+13)(x_1-15)$$
$$f_2(x_2)=0.01\,(x_2+15)(x_2+1)(x_2-8)$$
$$f_3(x_3)=0.01\,(x_3+9)(x_3-2)(x_3-9)$$
$$f_4(x_4)=0.01\,(x_4+11)(x_4+5)(x_4-9)$$
$$f_5(x_5)=0.01\,(x_5+9)(x_5-9)(x_5-10)$$

Different 5-rectangles were experimented with, all enclosing the maximum point.

(1) Region	List sizes	Bisections	Bounds
$x_1 \in [8.00, 9.00]$	67, 213	313	[22110.018, 25426.918]
$x_2 \in [-10.00, -9.00]$			
$x_3 \in [-5.00, -4.00]$			
$x_4 \in [3.00, 4.00]$			
$x_5 \in [-3.00, -2.00]$			

(2) Region	List sizes	Bisections	Bounds
$x_1 \in [8.50, 9.00]$	109, 121	231	[23603.552, 24936.918]
$x_2 \in [-10.00, -9.00]$			
$x_3 \in [-5.00, -4.00]$			
$x_4 \in [3.50, 4.00]$			
$x_5 \in [-3.00, -2.50]$			

(3) Region	List sizes	Bisections	Bounds
$x_1 \in [8.70, 8.80]$	2, 2	4	[24054.508, 24774.119]
$x_2 \in [-9.40, -9.30]$			
$x_3 \in [-4.60, -4.50]$			
$x_4 \in [3.50, 3.60]$			
$x_5 \in [-2.90, -2.80]$			

(4) Region	List sizes	Bisections	Bounds
$x_1 \in [8.75, 8.76]$	5, 3	9	[24400.432, 24431.565]
$x_2 \in [-9.36, -9.35]$			
$x_3 \in [-4.58, -4.57]$			
$x_4 \in [3.59, 3.60]$			
$x_5 \in [-2.85, -2.84]$			

We conclude that the algorithm given in this paper is a simple and reasonably efficient method for finding the range of values of a differentiable function over an n-dimensional rectangle $X^{(0)}$.

If $X^{(0)}$ is a small region, a single evaluation of the monotonicity test form (3.2) may suffice. The excess width will be of the order w^2 where w is the maximum width of $X^{(0)}$ in any coordinate direction. For large $X^{(0)}$, the algorithm given in this paper will produce arbitrarily sharp bounds on the range of values given sufficiently high machine precision (long enough word length in the representation of machine numbers). The stopping criterion in *step 19* of the algorithm allows the maximum accuracy available for the given machine precision which is used. As an alternative, we could replace *step 19* by a test which stops the computation when the improvement in the lower bound is less than some fixed amount and stop the computation sooner with a less sharp bound.

References

Alefeld, G., Herzberger, J.: Einführung in die Intervallrechnung (Böhling, K. H., Kulisch, U., Maurer, H., eds.). Mannheim-Wien-Zürich: Bibliographisches Institut 1974.

Hansen, E.: Global optimization using interval analysis — the multi-dimensional case. Numer. Math. *34*, 247—270 (1980).

Moore, R. E.: On computing the range of values of a rational function of n variables over a bounded region. Computing *16*, 1—15 (1976).

Moore, R. E.: Methods and applications of interval analysis. SIAM Publications *1979*.

Skelboe, S.: Computation of rational functions. BIT *14*, 87—95 (1974).

Professor R. E. Moore,
Shen Zuhe, and
N. S. Asaithambi
Computer Science Department
University of Wisconsin
1210 West Dayton Street
Madison, WI 53706, U.S.A.