BIT 14 (1974), 87–95

COMPUTATION OF RATIONAL INTERVAL FUNCTIONS

STIG SKELBOE

Abstract.

This paper presents a general algorithm for computing interval expressions. The strategy is characterized by a subdivision of the argument intervals of the expression and a recomputation of the expression with these new intervals. The precision of the result is limited only by the actual computer.

1. Definitions and presentation of the problem.

The width of the interval $X=[\underline{x},\overline{x}]$ is defined by $w(X)=\overline{x}-\underline{x}$. The centre of an interval $X=[\underline{x},\overline{x}]$ is computed as $c=(\overline{x}+\underline{x})/2$. If a real variable t occurs in an interval expression, then it is interpreted as the interval $T=[t,t]$.

Let $f(x_1,x_2,\ldots,x_n)$ be a rational function of the real variables $x_1,x_2,\ldots,x_n$. The expression f contains a finite number of arithmetic operations. A rational interval function $F(X_1,X_2,\ldots,X_n)$ is derived from f by replacing the real variabels $x_1,x_2,\ldots,x_n$ by the intervals $X_1,X_2,\ldots,X_n$ and by replacing the arithmetic operations by the equivalent interval operations; f is called the real restriction of F. In the following we assume $f(x_1,x_2,\ldots,x_n)$ to be bounded for $x_i \in X_i$, $i=1,2,\ldots,n$.

In this paper we are interested in determining the function $\overline{F}(X_1,X_2,\ldots,X_n)$ defined by

$$\overline{F}(X_1,X_2,\ldots,X_n) = \{f(x_1,x_2,\ldots,x_n) \mid x_i \in X_i,\ i=1,2,\ldots,n\}$$

Note that the value $\overline{F}$ is not the result of a direct computation of $F(X_1,X_2,\ldots,X_n)$. If X_i occurs m_i times in the expression $F(X_1,X_2,\ldots,X_n)$, then direct application of interval arithmetic will give:

$$\begin{aligned} &F(X_1,X_2,\ldots,X_n) \\ &\quad = \{h(x_{11},\ldots,x_{1m_1},x_{21},\ldots,x_{2m_2},\ldots,x_{n1},\ldots,x_{nm_n}) \mid x_{ij} \in X_i\,, \\ &\qquad i=1,2,\ldots,n \text{ and } j=1,2,\ldots,m_i\} \end{aligned}$$

where the function h is derived from f by replacing the variable x_i, which occurs m_i times, by the set of variables $x_{i1},x_{i2},\ldots,x_{im_i}$ where each variable occurs just once. Clearly we have

$$\overline{F}(X_1,X_2,\ldots,X_n) \subseteq F(X_1,X_2,\ldots,X_n)\,.$$

Received May 7, 1973. Revised October 27, 1973.

2. Theorems.

Theorem 1 and 2 below are found in Moore [1] and Hansen [2] respectively. To the best of our knowledge theorems 3 and 4 are new.

THEOREM 1. *If* $F(X_1, X_2, \ldots, X_n)$ *is a rational interval expression as defined in the preceding section, then*

$$X_1' \subset X_1, X_2' \subset X_2, \ldots, X_n' \subset X_n \Rightarrow F(X_1', X_2', \ldots, X_n') \subset F(X_1, X_2, \ldots, X_n)$$

for every set of interval numbers $X_1, X_2, \ldots, X_n$ *for which the interval arithmetic operations in* F *are defined.*

Let $f(x_1, x_2, \ldots, x_n)$ be a rational function as defined in the previous section. Rewrite f as follows:

$$f(x_1, x_2, \ldots, x_n) = f(c_1, c_2, \ldots, c_n) + g(x_1 - c_1, x_2 - c_2, \ldots, x_n - c_n) ,$$

where c_i is the centre of the interval X_i.

If an interval function $F(X_1, X_2, \ldots, X_n)$ is derived from the righthand side of the above equation, then the interval function is said to be in centred form.

THEOREM 2. *Let* $F(X_1, X_2, \ldots, X_n)$ *be in centred form. There exists a positive real number* k *such that*

$$F(X_1, X_2, \ldots, X_n) = \bar{F}(X_1, X_2, \ldots, X_n) + E$$

where $w(E) \leqq k \max_i (w(X_i)^2)$, $i = 1, 2, \ldots, n$ *and* $0 \in E$.

THEOREM 3. *Let* $F(X_1, \ldots, X_j, \ldots, X_n)$ *be a rational interval function as defined in section one. If* X_j *occurs only once in the function* F *and*

$$X_j = \bigcup_{i=1}^{N} X_j^{(i)}$$

then

$$F(X_1, \ldots, X_j, \ldots, X_n) = \bigcup_{i=1}^{N} F(X_1, \ldots, X_j^{(i)}, \ldots, X_n)$$

for every set of interval numbers for which the interval operations in F *are defined.*

PROOF. As $F(X_1, X_2, \ldots, X_n)$ is computed directly by means of interval arithmetic we can, according to section one, rewrite $F(X_1, X_2, \ldots, X_n)$ in the following way:

$$F(X_1,\ldots,X_j,\ldots,X_n) = H(X_{11},\ldots,X_{1m_1},\ldots,X_j,\ldots,X_{n1},\ldots,X_{nm_n})$$

where the value of $X_{ik} = X_i$, $i = 1,\ldots,j-1,j+1,\ldots,n$ and $k = 1,2,\ldots m_i$. Since each variable in H occurs just once we have:

$$H(X_{11},\ldots,X_{nm_n}) = \bar{H}(X_{11},\ldots,X_{nm_n}) .$$

From theorem 1 we know that

$$H(X_{11},\ldots,X_j,\ldots,X_{nm_n}) \supseteq \bigcup_{i=1}^{N} H(X_{11},\ldots,X_j^{(i)},\ldots,X_{nm_n}) .$$

In addition, we have

$$\bar{H}(X_{11},\ldots,X_j,\ldots,X_{nm_n}) \subseteq \bigcup_{i=1}^{N} H(X_{11},\ldots,X_j^{(i)},\ldots,X_{nm_n})$$

which implies

$$H(X_{11},\ldots,X_j,\ldots,X_{nm_n}) = \bigcup_{i=1}^{N} H(X_{11},\ldots,X_j^{(i)},\ldots,X_{nm_n})$$

thus completing the proof.

THEOREM 4. *Let $F(X_1,X_2,\ldots,X_n)$ be a rational interval function written in centred form. Each of the variables $X_{p+1},\ldots,X_n$ occurs just once in the function F.*

Subdivide each of the intervals $X_1,\ldots,X_p$ so that

$$X_i = \bigcup_{j=1}^{N} X_i^{(j)} \text{ with } w(X_i^{(j)}) = \frac{1}{N} w(X_i) .$$

There is a positive number k such that

$$\bigcup_{i_1=1}^{N} \cdots \bigcup_{i_p=1}^{N} F(X_1^{(i_1)},\ldots,X_p^{(i_p)},X_{p+1},\ldots,X_n) = \bar{F}(X_1,X_2,\ldots,X_n)+E_N$$

where

$$w(E_N) \leqq \frac{k}{N^2} \max_i \bigl(w(X_i)^2\bigr), \quad i = 1,2,\ldots,n \text{ and } 0 \in E_N .$$

PROOF. The intervals X_{p+1} through X_n are subdivided like the p first ones. From theorem 2 we know that

$$F(X_1^{(i_1)},\ldots,X_n^{(i_n)}) = \bar{F}(X_1^{(i_1)},\ldots,X_n^{(i_n)})+E_{i_1,\ldots,i_n}$$

where

$$w(E_{i_1,\ldots,i_n}) \leqq \frac{k_1}{N^2} \max_i \big(w(X_i)^2\big), \quad i=1,2,\ldots,n \text{ and } 0 \in E_{i_1,\ldots,i_n}$$

$$\bigcup_{i_1=1}^{N} \cdots \bigcup_{i_n=1}^{N} F(X_1^{(i_1)},\ldots,X_n^{(i_n)}) = \overline{F}(X_1,\ldots,X_n) + \bigcup_{i_1=1}^{N} \cdots \bigcup_{i_n=1}^{N} E_{i_1,\ldots,i_n} = \overline{F}(X_1,\ldots,X_n) + E_N' .$$

Hence

$$w(E_N') \leqq \frac{2k_1}{N^2} \max_i \big(w(X_i)^2\big) ,$$

since

$$0 \in E_{i_1,\ldots,i_n} \quad (i=1,2,\ldots,n) .$$

From theorem 3 we obtain

$$F(X_1^{(i_1)},\ldots,X_{n-1}^{(i_{n-1})},X_n) = \bigcup_{i_n=1}^{N} F(X_1^{(i_1)},\ldots,X_n^{(i_n)}) ,$$

$$F(X_1^{(i_1)},\ldots,X_{n-2}^{(i_{n-2})},X_{n-1},X_n) = \bigcup_{i_{n-1}=1}^{N} F(X_1^{(i_1)},\ldots,X_{n-1}^{(i_{n-1})},X_n)$$

and analogously for the rest of the $n-p$ variables with just one occurrence.

Hence

$$F(X_1^{(i_1)},\ldots,X_p^{(i_p)},X_{p+1},\ldots,X_n) = \bigcup_{i_{p+1}=1}^{N} \cdots \bigcup_{i_n=1}^{N} F(X_1^{(i_1)},\ldots,X_p^{(i_p)},X_{p+1}^{(i_{p+1})},\ldots,X_n^{(i_n)})$$

and

$$\bigcup_{i_1=1}^{N} \cdots \bigcup_{i_p=1}^{N} F(X_1^{(i_1)},\ldots,X_p^{(i_p)},X_{p+1},\ldots,X_n) = \bigcup_{i_1=1}^{N} \cdots \bigcup_{i_n=1}^{N} F(X_1^{(i_1)},\ldots,X_n^{(i_n)})$$

which implies $E_N = E_N'$ and completes the proof.

REMARK. Let $[\underline{y}_N,\bar{y}_N]=\bar{F}(X_1,X_2,\ldots,X_n)+E_N$. The following bounds are a direct result of theorem 4:

$$-\underline{e}_N = \underline{y}-\underline{y}_N \leqq \frac{k}{N^2}\max_i\left(w(X_i)^2\right) \text{ and } \bar{e}_N = \bar{y}_N-\bar{y} \leqq \frac{k}{N^2}\max_i\left(w(X_i)^2\right)$$

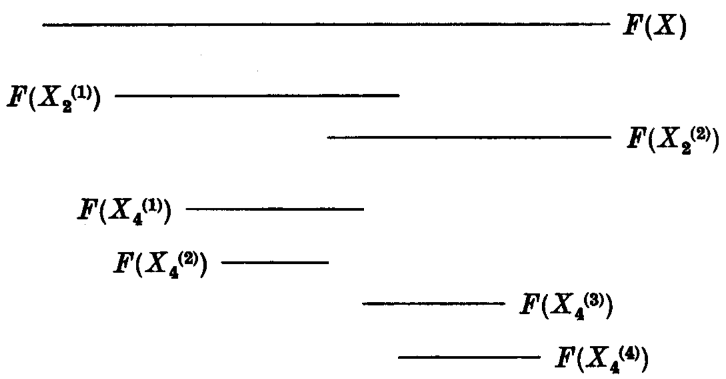
where $i=1,2,\ldots,n$, $[\underline{y},\bar{y}]=\bar{F}(X_1,X_2,\ldots,X_n)$, and $[\underline{e}_N,\bar{e}_N]=E_N$.

3. The algorithm.

From theorem 4 we know that $\bar{F}(X_1,X_2,\ldots,X_n)$ can be computed to any desired precision by means of $F(X_1,X_2,\ldots,X_n)$ by letting $N\to\infty$ in the subdivisions of the intervals.

A sequence of evaluations with the subdivisions $N,2N,4N,8N,\ldots$ gives a linear convergence of $w(E_N)\to 0$ (worst case: $w(E_{2N})=\frac{1}{4}w(E_N)$). An iterative scheme, which needs all the function values for each mesh, is not of practical use. However, it is possible to reduce the number of calculations.

Consider a rational interval function of one interval variable $F(X)$. In a subdivision of $X: X_N^{(1)},X_N^{(2)},\ldots,X_N^{(N)}$ we have $\bar{x}_N^{(i)}=\underline{x}_N^{(i+1)}$, $i=1,2,\ldots,N-1$. The width of the intervals is $w(X_N^{(i)})=(1/N)w(X)$. A sequence of computations might look like this:

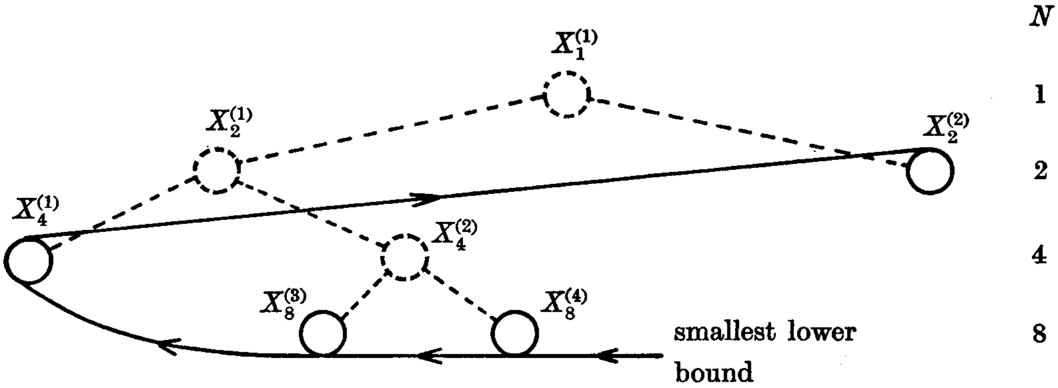


Consider the subdivision corresponding to $N=4$. In this case the lower bound of $\bigcup_{i=1}^{4}F(X_4^{(i)})$ can be found on the basis of $F(X_4^{(1)})$, $F(X_4^{(2)})$, and $F(X_2^{(2)})$, because $F(X_4^{(3)})\cup F(X_4^{(4)})\subset F(X_2^{(2)})$ (theorem 1).

This observation is the fundamental idea of the algorithm for the computation of $\bar{F}(X_1,X_2,\ldots,X_n)$. The upper and lower bounds are computed in the same way in all essentials. As an example the evaluation of the lower bound is described in more detail.

During the computation the lower bounds of F are linked together to form a linear list. The list is organized with the smallest lower bound as the first one. An example will explain the method of calculation.

With $N=8$ the situation might be the following:



With $[\underline{y}_N^{(i)}, \bar{y}_N^{(i)}] = F(X_N^{(i)})$ we have

$$\underline{y}_8^{(4)} \leqq \underline{y}_8^{(3)} \leqq \underline{y}_4^{(1)} \leqq \underline{y}_2^{(2)} .$$

The next step involves the division $X_8^{(4)} : X_8^{(4)} \to (X_{16}^{(7)}, X_{16}^{(8)})$ and computation of the corresponding values of the interval function. These values are placed in the list in proper order, and $F(X_8^{(4)})$, which is now superfluous, is discarded. Assume that $\underline{y}_{16}^{(7)} \leqq \underline{y}_{16}^{(8)}$; this gives two possibilities:

1) $\underline{y}_{16}^{(7)} \leqq \underline{y}_8^{(3)}$; we proceed with $N=32$
2) $\underline{y}_{16}^{(7)} > \underline{y}_8^{(3)}$.

In this case $X_8^{(3)}$ has to be subdivided, the corresponding functions computed, and the results ordered. This scheme is continued until the smallest lower bound corresponds to the subdivision $N=16$.

The iteration is stopped when the difference between two consecutive lower bounds $\underline{y}_N^{(i)}$ and $\underline{y}_{2N}^{(j)}$ is less than or equal to $\varepsilon \underline{y}_{2N}^{(j)}$: $|\underline{y}_{2N}^{(j)} - \underline{y}_N^{(i)}| \leqq \varepsilon |\underline{y}_{2N}^{(j)}|$. In all of the tested examples this criterion makes the relative error in the computed bounds less than ε.

For $\varepsilon = 0$ the termination criterion will be of the type described in Nickel and Ritter [3], and the conditions of numerical convergence of the algorithm will be fulfilled. This fact will be explained in greater detail.

In the case of rounded interval arithmetic theorem 1 is weakened to

$$X_1' \subset X_1, X_2' \subset X_2, \ldots, X_n' \subset X_n \Rightarrow F(X_1', X_2', \ldots, X_n') \subseteq F(X_1, X_2, \ldots, X_n)$$

but it still ensures that $\underline{e}_{2N} \geqq \underline{e}_N$ and $\bar{e}_{2N} \leqq \bar{e}_N$. This, in connection with the termination criterion and the remark after theorem 4, is the main condition for numerical convergence [3].

The algorithm is easily generalized to functions of more than one variable, and the present computer program is able to cope with functions of 30 variables. This restriction applies only to variables with multiple occurrences. In practice, however, this limit is lowered to 5–10 depending on the desired precision.

4. Test results.

The program has been tested with several functions and the following general tendency can be deduced from the tests. If the extremes of the interval function are assumed on the boundary of the argument set then the function is well behaved and fairly easy to compute. Otherwise, the evaluation requires a great amount of work and the more extremes of the interior of the argument set there are, the more work is required.

An example will demonstrate how the program works out in practice.

$$F(X_1, X_2, X_3) = \frac{X_1 + X_2}{X_1 - X_2} X_3$$

and in centred form

$$F_c(X_1, X_2, X_3) = Y_3 \left[\frac{c_1 + c_2}{c_1 - c_2} + \frac{2(c_1 Y_2 - c_2 Y_1)}{(c_1 - c_2 + Y_1 - Y_2)(c_1 - c_2)}\right]$$

where $c_1 = (\underline{x}_1 + \bar{x}_1)/2$, $c_2 = (\underline{x}_2 + \bar{x}_2)/2$,

$$Y_1 = X_1 - c_1, \quad Y_2 = X_2 - c_2, \text{ and } Y_3 = X_3$$

$$\bar{F}([1,2],[5,10],[2,3]) = [-7, -\tfrac{22}{9}]$$

while

$$F_c([1,2],[5,10],[2,3]) = [-7, -\tfrac{12}{9}].$$

In the case of two variables with multiple occurrences the graduation N corresponds to N^2 subdivisions.

```
PROCEDURE INTVFUNC(LONG REAL ARRAY XL,XH(*);LONG REAL YL,YH;
                    PROCEDURE FUNC;LONG REAL VALUE EPS;
                    INTEGER VALUE NM,NS,DIM);
  BEGIN
  COMMENT THE MEANING OF THE ARGUMENTS IS THE FOLLOWING:
    XL(I) AND XH(I) CONTAIN THE LOWER AND UPPER BOUND
      OF INTERVAL ARGUMENT NO. I.
    YL AND YH GIVE THE REFINED VALUE OF THE UPPER AND LOWER BOUND
      OF THE INTERVAL FUNCTION FUNC WITH THE ARGUMENTS XL AND XH.
    FUNC IS THE INTERVAL FUNCTION. FUNC MUST ACCEPT THE
      CALL FUNC(XL,XH,YL,YH).
    EPS IS THE EPSILON WHICH DETERMINES THE RELATIVE PRECISION OF
      THE RESULT YL AND YH.
    NM: VARIABLES XL,XH(1::NM) OCCUR MORE THAN ONCE IN FUNC.
    NS: VARIABLES XL,XH(NM+1::NM+NS) OCCUR JUST ONCE IN FUNC.
    DIM IS THE UPPER LIMIT OF THE WORKSPACE USED BY THE PROCEDURE.
      THE NECESSARY VALUE OF DIM CAN NOT BE PREDICTED. INITIAL
      GUESS: DIM:=4*(2**NM).
    MAXREAL IS THE LARGEST POSITIVE LONG REAL NUMBER PROVIDED
      BY THE IMPLEMENTATION.
    ***  VERSION 24/7 - 1973  ***;
  INTEGER MAX,FIRST,FREE,TEMP,RR,SS,L,COMB,LH;
  LONG REAL Y1,Y2,RO,RES,AEPS; LOGICAL STORE;
  INTEGER ARRAY NEXT,BACK,LEVEL(1::DIM); LONG REAL ARRAY VAL(1::DIM);
  LONG REAL ARRAY C(1::NM*3); LONG REAL ARRAY X1,X2(1::NM+NS);
  LONG REAL ARRAY LARG,HARG(1::DIM,1::NM);
    COMB:=ROUND(2**NM)-1; STORE:=FALSE;
    FOR M:=NM+1 STEP 1 UNTIL NM+NS DO
    BEGIN X1(M):=XL(M); X2(M):=XH(M) END;
    FOR K:=1,2 DO
      BEGIN
        MAX:=FIRST:=LEVEL(1):=1; FREE:=DIM-1; NEXT(1):=2;
        FOR P:=1 STEP 1 UNTIL FREE DO BACK(P):=P-1;
        VAL(2):=MAXREAL; NEXT(2):=0; TEMP:=DIM;
        FUNC(XL,XH,Y1,Y2); RES:=VAL(1):=IF K=1 THEN Y1 ELSE -Y2;
        FOR I:=1 STEP 1 UNTIL NM DO
        BEGIN LARG(1,I):=XL(I); HARG(1,I):=XH(I) END;
        AEPS:=ABS EPS*(1+ABS RES); RO:=ABS RES*2+AEPS+1;
        WHILE ABS(RES-RO)>AEPS DO
          BEGIN
            RO:=RES; MAX:=MAX+1;
            WHILE LEVEL(FIRST)<MAX DO
              BEGIN
                RR:=1; L:=LEVEL(FIRST)+1;
                FOR N:=1 STEP 1 UNTIL NM DO
                  BEGIN C(RR):=LARG(FIRST,N);
                  C(RR+2):=HARG(FIRST,N);
                  C(RR+1):=(C(RR)+C(RR+2))/2; RR:=RR+3 END;
                BACK(FIRST):=FREE; FREE:=FIRST;
                FIRST:=NEXT(FIRST);
                IF FIRST<0 THEN
                BEGIN WRITE("STORAGE LIST OVERFLOW"); GO TO OUT END;
                FOR R:=0 STEP 1 UNTIL COMB DO
                  BEGIN LH:=R; SS:=1;
                    FOR J:=1 STEP 1 UNTIL NM DO
                      BEGIN  RR:=LH REM 2+SS;
                      X1(J):=C(RR); X2(J):=C(RR+1);
                      SS:=SS+3; LH:=LH DIV 2 END;
                    FUNC(X1,X2,Y1,Y2); IF K=2 THEN Y1:=-Y2;
                    IF Y1<=VAL(FIRST) THEN
                    BEGIN STORE:=TRUE; NEXT(TEMP):=FIRST;
                    BACK(FIRST):=TEMP; FIRST:=TEMP END
                    ELSE BEGIN RR:=NEXT(FIRST);
                      WHILE Y1>VAL(RR) DO RR:=NEXT(RR);
                      IF RR>=0 THEN BEGIN STORE:=TRUE; SS:=BACK(RR);
                        NEXT(TEMP):=RR; BACK(TEMP):=SS;
                        BACK(RR):=TEMP; NEXT(SS):=TEMP END
                      END;
                    IF STORE THEN BEGIN STORE:=FALSE;
                      FOR Q:=1 STEP 1 UNTIL NM DO BEGIN
                      LARG(TEMP,Q):=X1(Q); HARG(TEMP,Q):=X2(Q) END;
                      VAL(TEMP):=Y1; LEVEL(TEMP):=L;
                      TEMP:=FREE; FREE:=BACK(FREE);
                      NEXT(FREE):=-1; VAL(FREE):=MAXREAL END
                  END R
              END LEVEL;
            RES:=VAL(FIRST); AEPS:=ABS EPS*(1+ABS RES);
          END BOUND;
        IF K=1 THEN YL:=RES ELSE YH:=-RES;
      END K;
OUT: END INTVFUNC;
```

Lower bound:

ε	relative error	N^2	number of function calls
10^{-2}	$< 10^{-14}$	4	5

Upper bound:

ε	relative error	N^2	number of function calls
10^{-2}	$0.2\cdot 10^{-2}$	64	17
10^{-4}	$0.3\cdot 10^{-4}$	$4\cdot 10^{3}$	29
10^{-6}	$0.09\cdot 10^{-6}$	10^{6}	45
10^{-8}	$0.2\cdot 10^{-8}$	$7\cdot 10^{7}$	57
10^{-10}	$0.5\cdot 10^{-10}$	$4\cdot 10^{9}$	69
10^{-12}	$0.1\cdot 10^{-12}$	10^{12}	85
10^{-14}	$< 10^{-14}$	$7\cdot 10^{13}$	97

5. The procedure INTVFUNC.

The procedure is written in ALGOL W for the IBM 370/165. In connection with this project the Assembler program INTV has been constructed. It performs the basic operations + − * and / in rounded interval arithmetic. INTV is written for the IBM system 360 and 370 and obeys the subroutine linkage conventions of FORTRAN G and H. A copy of INTV can be obtained from the Institute for Numerical Analysis, The Technical University of Denmark, DK 2800 Lyngby.

If TRIPLEX-ALGOL 60 [4] had been available at our computing center it would of course have been used instead of INTV.

Acknowledgement.

I wish to thank Mr. Kaj Madsen for many helpful discussions and suggestions. Further I want to thank the referee for several valuable comments.

REFERENCES

1. R. E. Moore, *Interval Analysis*, p. 11, Prentice–Hall, 1966.
2. E. R. Hansen, *Topics in interval Analysis*, p. 103, Oxford University Press, 1969.
3. K. Nickel and K. Ritter, *Termination criterion and numerical convergence*, SIAM J. Numer. Anal. 9 (1972), 277–283.
4. N. Apostolatos et al., *The algorithmic language Triplex-Algol 60*, Numerische Mathematik 11 (1968), 175–180.

NUMERISK INSTITUT
DANMARKS TEKNISKE HØJSKOLE
DK-2800 LYNGBY
DENMARK