INTERVAL OPERATIONS AND FUNCTIONS LIBRARY FOR FORTRAN 77 PROGRAMMING SYSTEM AND ITS PRACTICE USING

Vladimir N.Krishchuk, Nikolay M.Vasilega and Galina L.Kozina

On the basis of the INFFOR language a library of the interval operations and functions was created for the programming system Fortran 77 corresponding to the popular version 5.0 of the compiler MS Fortran. The worked out library was named INTFOR.LIB and supports the interval operation in different types of formation and comparison of the interval numbers as well as the functions of the interval arithmetics and interval spreading of the elementary functions. The library may be used in a way which is conventional for the programming system Fortran 77 of the personal computer IBM AT with a mathematic co-processor.

БИБЛИОТЕКА ИНТЕРВАЛЬНЫХ ОПЕРАЦИЙ И ФУНКЦИЙ ДЛЯ СИСТЕМЫ ПРОГРАММИРОВАНИЯ ФОРТРАН 77 И ПРАКТИКА ЕЕ ИСПОЛЬЗОВАНИЯ

В.Н.Крищук, Н.М.Василега, Г.Л.Козина

На основе языка INFFOR создана библиотека интервальных операций и функций для системы программирования Фортран 77, соответствующей популярной версии 5.0 компилятора MS Fortran. Разработанная библиотека получила название INTFOR.LIB и поддерживает интервальные операции по различным видам формирования и сравнения интервальных чисел, функции интервальной арифметики и интервальные расширения элементарных функций. Библиотека может использоваться стандартным для системы программирования Фортран 77 способом на ПЭВМ типа IBM AT, имеющей математический сопроцессор.

IONS
MING
G

ι L.Kozina

erval oper-
Fortran 77
rtran. The
erval oper-
al numbers
)reading of
ich is con-
computer

ІЦИЙ
?ОВАНИЯ
ВАНИЯ

зальных
гран 77,
Fortran.
В и под-
форми-
ерваль-
х функ-
і систе-
па IBM

The authors have used the package of application programs proposed by Pankova [1] that implements the algorithmic language INFFOR extending possibilities of FORTRAN 4 and ensuring the support of interval calculations. Its application allowed one solving problems of strength characteristics of radioelectronic means constructing under external mechanical actions to the computer ES-1045. The package is maintainable and does not require much time to pass from real parameters to interval ones as well as a substantial changes of software available. However, INFFOR is ES oriented. In this case subprograms of interval analysis are implemented on the ASSEMBLER language, and the preprocessor for the command language is implemented on PL/1. Taking into account an extending using of workstations based on PC while constructing radioelectronic means, the necessity arises of solving problems of analysis of dynamical characteristics of constructions using PC's of the type IBM PC XT/AT and, therefore, the necessity of transfer from the computer ES and adaptation of a large-scale software implemented in FORTRAN.

In this connection, a library of interval operations and functions for the program system FORTRAN 77 corresponding to the popular 5.0 version of the compiler MS FORTRAN has been created on the basis of the INFFOR language for PC. This library was named INTFOR.LIB; it supports interval operations of various kind of forming and comparison of interval numbers, functions of interval arithmetic and interval extensions of elementary functions [1,2,3].

Library description. A detailed description of the syntax of INFFOR can be found in [4]. Here, we consider its characteristic features only. The alphabet of the language, besides the standard alphabet of FORTRAN 77, contains the symbol ";" (the semicolon). This symbol is used in recording interval constants as the delimiter of the integer and fractional parts of a numbers. To describe the interval type, only the default principle is used. Identifiers denoting interval variables and arrays must begin with one of letters, A, B, C, D, E, F, G, H but the use of the letter G is not recommended since it is reserved for library functions.

An undeniable advantage of work [1] holds as for the using of a preprocessor to converse the source code from INFFOR language (the file with name *.INF) into corresponding program module in FORTRAN 77 (file *.FOR). The preprocessor INFOR.EXE is implemented in FORTRAN 77 and has been modified according to the features of a PC architecture

and to the work with symbolic data as well as the I/O organization. The preprocessor executes the following operations:

(1) — specifies implicit description of the interval type by adding the operator IMPLICIT REAL 8(A-H) in program modules;

(2) — converses records of interval constants and arithmetical operations over intervals into corresponding records using library functions as well as records of printing operations for interval variables;

(3) — makes syntactic control of source code operators in INFOR language.

As the result of the preprocessor work, two files: *.FOR and *.FTR arise. The first file contains the code in FORTRAN 77, and the second, in addition, contains information on syntactic errors. Further work with the file *.FOR (compilation) is performed according to the rules of PC FORTRAN 77. Notice that the preprocessor does not process operators that contain interval operations if they have continuation lines.

Interval operations and functions are implemented in ASSEMBLER language in the form of program modules of function type. In order to keep succession in designs, function names correspond to that adopted in [1,4]. We give here a brief description of interval operations and functions:

(1) GP(R1,R2) is a generation of interval number $[R1,R2]$;

(2) GP(I1,I2) is the generation of interval number $[I1,I2]$;

(3) GR(R) is the generation of interval number $[R,R]$;

(4) GI(I) is the generation of interval number $[I,I]$;

(5) RI(I) is the conversion of an integer into a real number;

(6) RL(G) is the allocation of the lower bound of the interval number G: $\underline{G}$;

(7) RM(G) is the allocation of the upper bound of the interval number G: $\overline{G}$;

(8) GRL(G) is the generation of an interval number on the lower bound of G: $[\underline{G}, \underline{G}]$;

(9) GRM(G) is the generation of an interval number on the upper bound of G: $[\overline{G}, \overline{G}]$;

(10) GHL(G) is the allocation of the lower half of G: $[\underline{G}, (\underline{G} + \overline{G})/2]$;

(11) GHM(G) is the allocation of the upper half of G: $[(\underline{G} + \overline{G})/2, \overline{G}]$;

(12) GAL(G) is the generation of an interval number on the point nearest to zero of G: $[\underline{G}, \underline{G}]$, $[0, 0]$ or $[\overline{G}, \overline{G}]$;

(13) ISG(G1,(ify the si; −1 if G1· 0 if the i +1 if G1:

(14) ISN(G) t the sign (−1 if G< 0 if 0 do(+1 if G>

(15) IMP(G1, specify a: −1 if G1 0 if the i G2; +1 if the

(16) GUN(G1 intersect:

(17) GIS(G1,(

(18) GC(R1,F R2 > 0 : [[R1/R2]

(19) GT(R1,F [[R1 * R2

(20) GN(G) i the nota

(21) GA(G1,(cording 1

(22) GS(G1,(accordin

(23) GM(G1, accordin

(24) GD(G1,(bers G1 does not

(25) GW(G,I integer]

(26) RLP(G,

The left margin contains fragments of text (cut off):

INA

rganization. The

type by adding
am modules;
arithmetical op-
ds using library
ions for interval

ators in INFOR

OR and *.FTR
and the second,
ther work with
he rules of PC
ocess operators
lines.

ASSEMBLER
e. In order to
hat adopted in
and functions:

,R2];
I2];

nber;
terval number

interval num-

on the lower

on the upper

$(\underline{G} + \overline{G})/2]$;
$\underline{+} \overline{G})/2, \overline{G}]$;
on the point

(13) ISG(G1,G2) the obtaining of an integer constant in order to specify the sign of the difference G1 and G2:
-1 if G1$<$ G2;
0 if the intersection is non-empty;
$+1$ if G1$>$ G2;

(14) ISN(G) the obtaining of an integer constant in order to specify the sign of the interval number G:
-1 if G$<$ 0;
0 if 0 does not belong to G;
$+1$ if G$>$ 0;

(15) IMP(G1,G2) the obtaining of an integer constant in order to specify an imbedding of G1 into G2:
-1 if G1 is imbedded into G2;
0 if the intersection is non-empty, but G1 is not imbedded into G2;
$+1$ if the intersection is empty;

(16) GUN(G1,G2) is the representation by the outer interval of the intersection of G1 and G2;

(17) GIS(G1,G2) is the intersection of G1 and G2 if it is not empty;

(18) GC(R1,R2) is the division of the real number R1 $\geq$ 0 and R2 $>$ 0 :
$[[R1/R2], [\overline{R1/R2}]]$;

(19) GT(R1,R2) is the multiplication of the real number R1 and R2:
$[[R1 * R2], [\overline{R1 * R2}]]$;

(20) GN(G) is the sign change of the interval number G according to the notation $-G$;

(21) GA(G1,G2) is the sum of the interval numbers G1 and G2 according to the notation: G1 $+$ G2;

(22) GS(G1,G2) is the difference of the interval numbers G1 and G2 according to the notation: G1 $-$ G2;

(23) GM(G1,G2) is the product of the interval numbers G1 and G2 according to the notation: G1 $*$ G2;

(24) GD(G1,G2) is the quotient from the division of the interval numbers G1 and G2 according to the notation: G1/G2 provided 0 does not belong to G2;

(25) GW(G,I) is the raising of the interval numbers G to a nonnegative integer power I according to the notation: G^I;

(26) RLP(G,I) is the computing of the lower boundary of G multiple

V.N.KRISHCHUK, N.M.VASILEGA, G.L.KOZINA

to 10^{-I};

(27) RMP(G,I) is the computing of the upper boundary of G multiple to 10^{-I};

(28) GABS(G) is the computing of the absolute value of G;

(29) GMIN(G1,G2) is the computing of the minimum of G1 and G2;

(30) GMAX(G1,G2) is the computing of the maximum of G1 and G2;

(31) GSQRT(G) is the computing of the square root of G;

(32) GSIN(G) is the computing of the sinus of G;

(33) GCOS(G) is the computing of the cosinus of G;

(34) GLOG(G) is the computing of the logarithm of G;

(35) GEXP(G) is the computing of the exponent of G.

Program implementation of the functions GPI(I1,I2),GI(I), RI(I), GAL(G), GLOG(G), RLP(G,I), and RMP(G,I) in FORTRAN 4 is taken from [1].

Corresponding object modules are put in the library INTFOR.LIB. One can use the library in a standard way for the software complex FORTRAN 77 in the personal computer of IBM PC XT/AT type with mathematical processor [5]. The implementation of interval operations and functions in ASSEMBLER language [6] contributes to a speed raise of the solution of problems and to the guarantee of the comfortable interface with FORTRAN 77 programs.

The collection of interval operations and functions supported by the library INTFOR.LIB, becomes wider, in particular, the complex interval arithmetic is implemented [7,8].

Practical application. In Zaporozhsky Institute on Machine Engineering, a software complex for analysis and guarantee of reliable construction of airborne radioelectronic means under long vibrational actions aimed to the exploitation on computers of IBM PC AT type. Most of computing programs of this package is implemented with the use of the programming system FORTRAN 77. For the support of interval operations and functions, the library INTFOR.LIB is used.

The software complex is intended for the analysis of dynamical characteristics of carrying constructions and printed circuit-cards of radioelectronic means under vibrational actions as well as for estimating of their reliability indices according to the fatigue strength [9,10].

Applied pr
electronical m
structed on th
actions can be
erations or sin
brational load:
rate change is
initial strains :

Initial data
given in the tr
supposed) anc
nological toler
action values
using interval
of interval nur

The reliabi
Sosnovsky [11]
ies by the auth
to develop anc
construction e
limit.

The softwa
to form an ac
data, browse
output data in
5.5 version of

1. Pankova, G.
   Frunze, 1980

2. Kalmykov, S.
   Nauka, Novo

3. Alefeld, G. a
   Press, New Y

4. Pankova, G.I
   putations on
   (1992), pp.86

Applied programs of analysis of mechanical processes in the radio-electronical means constructions implement mathematical models constructed on the basis of the finite element method. External mechanical actions can be given in the form of static loads, linear (centrifugal) accelerations or sinusoidal harmonic vibration. The possibility for giving vibrational loads in the form of cynematic or force action with programmed rate change is implemented. One can also to compute constructions with initial strains resulting the manufacture or the assembly of details.

Initial data describing the construction and external actions can be given in the traditional form (nominal values or values for the worst case supposed) and in the form taking into account constructive and technological tolerances on construction elements and tolerances on external action values (for example, a vibrational acceleration). This is done by using interval analysis tools and giving a priori information in the form of interval numbers.

The reliability computing methods are based on the works by L. A. Sosnovsky [11] and V. P. Kogaev [12], as well as on the experimental studies by the authors. The interval setting of the analysis problem permitted to develop and refine the method of computing of the fatigue strength of construction elements with account of gradual reducing of the endurance limit.

The software complex has a friendly interface allowing to the user to form an actual job in multiwindow mode, input and update input data, browse computing results in array and graphical representation, output data into printer. The software tools are implemented using the 5.5 version of TURBO-PASCAL.

References

1. Pankova, G.D. *Program package for validating computations on ES computer.* Frunze, 1980, deposited in VINITI 16.08.80, 2392-80 (in Russian).

2. Kalmykov, S.A., Shokin, Yu.I. and Yuldashev, Z.Kh. *Methods of interval analysis.* Nauka, Novosibirsk, 1986 (in Russian).

3. Alefeld, G. and Herzberger, J. *Introduction to interval computations.* Academic Press, New York, 1983.

4. Pankova, G.D. *Programming support and algorithmic language for validating computations on ES (IBM 360/370) series computer.* Interval computations 1 (3) (1992), pp.86-95 (in Russian).

5. Brich, Z.S., Kapilevich, D.V. and Kletskova, N.A. *FORTRAN-77 for PC ES: handbook.* Finansy i statistika, Moscow, 1991 (in Russian).

6. Scanlon, L.J. *IBM-PC assembly language: a guide for programmers.* Robert J. Brady Co., Bowie, MD, 1983.

7. Rokne, J. and Lankaster, P. *Complex interval arithmetic.* Comm. ACM **14** (1971), pp.111–112.

8. Rokne, J and Lankaster, P. *Complex interval arithmetic. Algorithm 86.* Comput. J. **18** (1975), pp.83–86.

9. Krishchuk, V.N. and Vasilega, N.M. *Program package for analysis and ensurance of reliability of failure-free on-board radio-electronic devices under prolonged vibration.* In: "International seminar "Modelling and control of quality in the problems of securing of reliability of electronic systems", Shaulyai, June 1-5, 1992", Shaulyai, 1992, pp. 28–29 (in Russian).

10. Krishchuk, V.N. and Vasilega, N.M. *Methods for study of failure-free data of radio-electronic on-board devices under prolonged vibrational actions.* In: "International conference "Methods and means of estimation and increasing of reliability of devices, equipment and systems", Penza, February 20–22, 1992", Penza, 1992, pp.46–48 (in Russian).

11. Sosnovsky, L.A. *Statistical mechanics of fatigue destruction.* Nauka i tekhnika, Moscow, 1987 (in Russian).

12. Kogaev, V.P. *Calculations of strength under alternating strains.* Mashinostroenie, Moscow, 1977 (in Russian).

ALGORITH
FOR L
W

In the prese
design is cons
the transfer m
based on the
Kharitonov's t

АЛГОРИ
НАДЕЖ

ИН'

В работе
ния, устойчи
ства описанс
Предлагаем
стических п

In modern cc
a provision of t
bustness. The s
response of the

© V.G.Kryms