数 値 解 析 17－3
（１９８６．７．４）

区間演算機能を持つ PASCAL-SC について

野寺 隆
（慶應義塾大学理工学部）

パーソナル・コンピータで利用できる区間演算機能を持つ高級言語PASCAL SC の使用法について述べる。この言語は教育面においても価値があるだけでなく，数値計算のあらゆる分野で計算結果（解）を評価する強力な道具となる。最後に，区間演算を用いた数値例について述べる。

Interval Arithmetic of Pascal-SC

Takashi Nodera
Faculty of Science and Technology
Keio University

This guide introduces the use of the high level language of PASCAL-SC with interval arithmetic routines on a personal computer. This language is found to be not only a valuable for the aspect of instruction, but also a powerful tool for evaluating the solution of a wide class of numerical computation in various area. At last, numerical examples, using interval arithmetic, are given.

1.はじめに

物理過程を記述する数学モデルの多くのものは，数学的な解析によって解くことができないので，一連の代数的な操作，即ち，数値計算を用いて解を求めることになる。しかし，数値計算により求めた計算結果が，厳密解と比較して，はたしてどの程度まで正しい精度で計算されているかを判断することは，非常に困難なことである。それは，今日の多くの電子計算機が採用している浮動小数点演算によるためである。従来の浮動小数点演算は，数学で厳密に成立する種々の法則（加減乗算の結合，分配法則など）を満足しないので，この演算方式により得られた計算結果だけでは，「得られた解がどの程度まで正確か。」という解の精度の保証が得られないからである。その主な理由を上げると，次に述べる点にある。

（1）初期データの不正確性。
（2）有限桁の浮動小数点演算により混入する丸め誤差や桁落ちの影響。
（3）無限級数や反復計算を有限のステップで打ち切るため。

上記の原因を打開するには，これらの事柄の影響を精密に調査し，「正しい解は，必ずこの区間内に存在する。」ことを保証するような演算方式を用いる必要がある。即ち，データを［下限，上限］の組（ペア）で表現し，区間と区間の演算によって求める解の存在区間を計算することである。これが，区間演算と呼ばれているものである。しかし，浮動小数点演算の計算量と比較すると，区間演算の計算量はあまりにも多くかかり過ぎることなど，さまざまな問題点もあり，区間演算方式に対する不信感を懐く人も少なからずいるのである。

区間演算用のパッケージには，従来，数種類のものがあり特定の利用者には大型汎用計算機で使用されてきた。最近，この区間演算パッケージとして，IBM の高精度演算サブルーチンACRITHが登場し，IBM 4361や S／370 で使用可能になった。

近年，マイクロ・コンピュータの大々的な普及により，身近なパーソナル・コンピュータで，手軽に数値計算を楽しむことができるようになった。また，プログラミング言語も従来のBASIC だけでなくTurbo Pascalなども現れてきたのだが，その中で区間演算を手軽に利用できるものが現れてきた。それが，PASCAL-SC である。

PASCALは，1968年に，ETH-Zurich大学のNiklaus Wirth 教授により開発されたもので，良いプログラミング形式を理解しやすいように設計された構造化プログラミング言語である。PASCAL-SC は，このオリジナルPASCALに基づいて製作されたシステムで，特に，区間演算をするためのサブルーチンを兼ね備えている。

プログラミング言語としてのPASCAL（およびPASCAL-SC ）の原理的なメリットについて述べると，次のようになる。

（1）透明性（プログラムの意図の明瞭性，即ち，プログラムが簡単に読め，理解できること）．
（2）安全性（プログラミングの犯しやすい多くの誤りが，コンピュータで検出可能なこと）．
（3）実行効率（プログラムが実時間で動くこと）．

さらに，PASCAL（PASCAL-SC ）が持つ利点の１つは，制御データの構造的な使い安さであり，そのデータ構造は，関連したデータ項目を一緒にグループ分けすることが可能な点である。

PASCAL（PASCAL-SC ）プログラムは，一般に，次のような基本構成になっている。

（ｉ）　＜プログラムの表題部＞
（ii）　＜全域的な名札宣言部＞
（iii）　＜全域的な定数定義部＞
（iv）　＜全域的な型定義部＞
（ｖ）　＜全域的な変数宣言部＞
（vi）　＜全域的な手続きと関数の宣言部＞
（vii）　＜プログラムの本文の部＞

（ｉ）は，頭書き（heading ）とブロックと呼ばれる本体に分かれる。（ii）は，ブロックの中で定義する全ての名札（ラベル）を並べて書く。（iii）定数に対する同義語を定義する。後で使用する定数名を書く。（iv）データ型の定義をする。（ｖ）変数を定義する。（vi）従属的なプログラム部分を書く。（vii）このプログラムで実行しようとする動作の指定をする。

2.PSACAL SC の概要

PSACAL-SC は，Wirth 教授がオリジナルなPASCAL（1968年版）の概念を改良した改訂PASCAL（1973年版）に基づいて製作されている。特に，数値計算において重要となるオペレータの概念に加え，次のようないくつかの拡張された機能を持っている。

（a）　Linkage with pretranslated subroutines
（b）　ELSE exit in CASE statements
（c）　A freer sequence of definitions and declartions
（d）　Dynamic file correspondence {only in the CP／M^m version}
（e）　Function results of arbitrary type
（f）　Underscore in identifiers
（g）　Implementation of many of the operating system functions of PASCAL

PSACAL-SC システムは，Z80 またはZ80Aのプロセッサのもとで，CP／M^m (version 2.0 以上）のオペレーティン

グシステム上で

の作業領域（RAM）と，2つのフロッピーディスク装置も必要である。当然のことながら，PC9801（NEC）でこのシステムを起動するためには，Z80 cardが必要である。

PASCAL-SC（Z80，CP／M™ 版）は，下記のソフトウェア・ハウスで入手できる。

〔住所〕 FBSoftware
135 N. Prospect Ave.
Madison, Wisconsion 53705
U.S.A.

〔価格〕 $50.00 （マニュアル1冊，システムは2枚のフロッピーディスク（single density）で提供される。）

また，68000 用のPASCAL-SC も（U.S.A で）存在しているのだが，入手方法は不明である。

3.PASCAL-SC の区間演算パッケージ

標準関数パッケージには，実数演算用，実区間演算用，複素数演算複素，区間複素数演算用などがある。

3.1 標準区間演算パッケージ

区間演算パッケージを利用するには，次のような型宣言を行う必要がある。

```
TYPE INTERVAL = RECORDED INF, SUP: REAL END ;
```

だだし，INF は区間の下限，SUP は区間の上限を示すものである。区間演算パッケージは次の6つの部分から構成されている。

（1）Transfer functions
（2）Comparisons
（3）Lattice operations
（4）Arithmetic operations
（5）Standard functions
（6）Input／output procedures

標準関数を除いて，すべて部分は，PASCAL-SC で製作されている。各部分の構成について略記すると次のようになる。

(1) Transfer functions

```
FUNCTION INTPT (RA : REAL) : INTERVAL ;
    {one real expression ⇒ interval }
FUNCTION INTVAL (RA, RB : REAL) : INTERVAL ;
    {two real expression ⇒ interval }
FUNCTION IINF ( A : INTERVAL) : INTERVAL ;
    {interval⇒ lower limit}
FUNCTION ISUP ( A : INTERVAL) : INTERVAL ;
    {interval⇒upper limit }
```

(2) Comparisons

```
OPERATOR <=  (A, B : INTERVAL) RES : BOOLEAN ;
    {<= は，⊂ を意味する。}
OPERATOR >=  (A, B : INTERVAL) RES : BOOLEAN ;
    {>= は，⊃ を意味する。}
OPERATOR IN (RA : REAL ; B : INTERVAL)
                                RES : BOOLEAN ;
    { IN は，包含（inclusion）を意味する。}
OPERATOR IN (KA : INTEGER ; B : INTERVAL)
                                RES : BOOLEAN ;
OPERATOR ><  (A, B : INTERVAL) RES : BOOLEAN ;
    { >< は，もし2つの区間が互いに素（dis-
    joint）ならば，空の交わりを持つことを意味する。}
```

(3) Lattice operations

```
OPERATOR ++ (A, B : INTERVAL)  RES : INTERVAL ;
    {++は，A とB を含む最も小さい凸包を示す。}
OPERATOR ** (A, B : INTERVAL)  RES : INTERVAL ;
    {**は，A とB の交わりを示す。}
```

(4) Arithmetic operations

```
OPERATOR +  (A : INTERVAL) RES : INTERVAL ;
OPERATOR -  (A : INTERVAL) RES : INTERVAL ;
OPERATOR +  (A, B : INTERVAL) RES : INTERVAL ;
OPERATOR +  (KA : INTEGER ; B : INTERVAL)
                                RES : INTERVAL ;
OPERATOR +  (A : INTERVAL ; KB : INTEGER )
                                RES : INTERVAL ;
OPERATOR -  (A, B : INTERVAL) RES : INTERVAL ;
OPERATOR -  (KA : INTEGER ; B : INTERVAL)
                                RES : INTERVAL ;
OPERATOR -  (A : INTERVAL ; KB : INTEGER )
                                RES : INTERVAL ;
OPERATOR *  (A, B : INTERVAL) RES : INTERVAL ;
OPERATOR *  (KA : INTEGER ; B : INTERVAL)
                                RES : INTERVAL ;
OPERATOR *  (A : INTERVAL ; KB : INTEGER )
                                RES : INTERVAL ;
OPERATOR /  (A, B : INTERVAL) RES : INTERVAL ;
OPERATOR /  (KA : INTEGER ; B : INTERVAL)
                                RES : INTERVAL ;
OPERATOR /  (A : INTERVAL ; KB : INTEGER )
                                RES : INTERVAL ;
```

マイナスの演算子（－）に対しては，次のようになる。

-A : = (-1) A = [-A.SUP, -A.INP]

(5) Standard functions

```
FUNCTION IABS (Y : INTERVAL) : REAL ;
```

```
    {IABS (Y ) : = max { | Y.INF | , | Y.SUP | }
                                        : 絶対値}
  FUNCTION ISQR  (Y : INTERVAL) : INTERVAL ;
    {ISQR (Y ) ⊃ y² ∈Y : 平方}
  FUNCTION ISQRT (Y : INTERVAL) : INTERVAL ;
    {ISQRT  (Y ) : 平方根}
  FUNCTION IEXP  (Y : INTERVAL) : INTERVAL ;
    {IEXP (y ) : 指数 (eのy乗の計算) }
  FUNCTION ILN   (Y : INTERVAL) : INTERVAL ;
    {ILN  (y ) : 自然対数}
  FUNCTION IARCTAN (Y : INTERVAL) : INTERVAL ;
    {IARCTAN  (y ) : 逆正接}
  FUNCTION ISIN  (Y : : INTERVAL) : INTERVAL ;
    {ISIN (Y ) : 正弦}
  FUNCTION ICOS  (Y : INTERVAL) : INTERVAL ;
    {ICOS (Y ) : 余弦}
```

(6) Input /output procedures

```
  PROCEDURE IREAD  (VAR F : TEXT ; VAR A : INTERVAL) ;
    {区間データの入力}
  PROCEDURE IWRITE (VAR F : TEXT ; A : INTERVAL) ;
    {区間データの出力}
```

3.2 区間ベクトル・行列演算パッケージ

区間行列計算パッケージを利用するには，次のようにする。

```
  CONST DIM =   ; {DIM is actual dimension of the
                   matrices and vectors used }
  TYPE  DIMTPE=1..DIM ;
        INTERVAL= RECORD INF, SUP : REAL END ;
        IVECTOR = ARRAY [DIMTYPE ] OF INTERVAL ;
        IMATRIX = ARRAY [DIMTYPE ] OF IVECTOR ;
```

これに加え，3.1 節で述べた標準区間演算パッケージも必要なことは明らかである。また，区間行列演算パッケージは，前述の標準区間演算パッケージと同様に 6 つの部分〔(1) ～ (6) 〕で構成されている。

4.PASCAL-SC のシステム及びライブラリーファイル

PSSCAL-SC システムは，55のファイルから構成されている。これらのファイルは，次の 8 つに分類される。

(1) Installation
(2) Compiler
(3) Executer
(4) Interpreter
(5) Run-time Error Handler
(6) External Libraries
(7) Demonstration Programs

(8) Files Generated during Compilation

各カテゴリーにどんなファイルが存在するかを述べることにする。

(1) Instalation

COPYSYS.SUB	Copies files needed from the CP/M^m system disk to the PASCL-SC system disk.

(2) Compiler

COMPILE.SUB	Drive the compilation process.
PARSER.KLP	Syntatic analysis (pass 1) .
SEMAN.KLP	Semantic analysis (pass 2) .
KLPMAP.KLP	Linking, code generation (pass 3) .
ERRLIST.KLP	Lists errors found during compilation.
SYMTAB.TXT	Symbol table (used by PARSER) .
LIBDIR.BIN	Library directory (used by KLPMAP) .

(3) Executer

XQP.COM	Initiates and manages execution of PASCAL-SC programs.
XQPC.COM	Sets all unspecified external file references to CON: , otherwise, the same as XQC.

(4) Interpreter

PINT00.RLC	Integer arithmetic.
PINT10.RLC	+ Floating-point arithmetic.
PINT11.RLC	+ Standard functions (no scalar product) .
PINT12.RLC	+ Scalar product (no standard function) .
PINT13.RLC	+ Scalar product and standard function.

(5) Run-time Error Handler

ERRLINE.KLP	Locates error location.
KLPERROR.RLC	Interpreter.
KLPERROR.TXT	List of KL/P (“Kaiserslautern PASCAL”) errors.

(6) External Libraries

DIMPAKET.SNT	Vector and matrix type declarations.
TYPEPAKET.SNT	Other type declarations (COMPLEX, INTERVAL) .
SFLIB.SRC	Standard function.
RPAKET.SNT	Additional standard function.
RPAKET.SNT	
RLIB.BIN	
CPAKET.SRC	Complex arithmetic.

CPAKET.SNT	
CDLIB.BIN	
IPAKET.SRC	Interval arithmetic
IPAKET.SNT	
ILIB.BIN	
MRPAKET.SRC	Subroutines for real vectors and
MRPAKET.SNT	matrix arithmetic.
MCPAKET.SRC	Subroutines for complex vector and
MCPAKET.SNT	matrix arithmetic.
MCPAKET.SRC	Subroutines for interval vector and
MIPAKET.SNT	matrix arithmetic.
SUMPAKET.SRC	Acculate sum of floating-point
SUMPAKET.SNT	numbers
SUMLIB.BIN	
LGLPAKET.SRC	Acculate solution of real and
	interval linear systems and
LGLPAKET.SNT	inversion of real and interval
	matrices.
LGLLIB.BIN	
EIGPAKET.SRC	Acculate calculation of eigenvalues
EIGPAKET.SNT	and eigenvectors of real matrices.
EIGLIB.BIN	
GINLIB.BIN	Floating-point matrix inversion (used only by LGL, EIG) .

(7) Demonstlation Program

デモンストレーション・プログラムの実行は次のようにする。

XQPC d:< name >

または

XQP d:< name > CON: CON:

ただし, < name >=FORMULA, FBLEIG, FBLGL, FBLINV, または FBLPOLY である。

FORMULA.KLP	Acculate evalution of arithmetic
	expression, with guranteed
FORMULA.HLP	inclusions of results.
FBLEIG.KLP	Acculate calculation of eigenvalues /vectors of symmetric matrices, with guaranteed inclusions of results.
FBLGL.KLP	Acculate solution of linear systems of equations, with guaranteed inclusions of results.
FBLINV.KLP	Acculate matrix inversion, with guaranteed inclusions of results.
FBLPOLY.KLP	Acculate calculation of values and roots of real polynomials, with guranteed inclusions of results.

(8) Files Generated during Compilation

PASCALで書かれたソースファイル <program.name>.SRC のコンパイルは, 次のようにすればよい。

SUBMIT COMPILE d:<program name>

または, 次の手順でコンパイルすればよい。ただし, d はディスク装置番号である。

XQP PASER d:<program name>.SRC, CON:, SYMTAB, SYNOUT

XQP SEMAN CON:, ERRMESS, SYNOUT, ZC

XQP KLPMAP CON:, ZC, LIBDIR, d:<program name>.LIN, d:<program name>.KLP

XQP ERRLIST d:<program name>.SRC, ERRMESS, LISTING

TYPE LISTING.TXT

次のファイルは, コンパイルを行った過程で生成される。

SYNOUT.BIN	Symbolic code produced by PARSER as input to SEMAN.
ZC.BIN	Intermediate code produced by SEMAN as input to KLPMAP.
ERRMESS.BIN	Code for errors found by PARSER and SEMAN as input to ERRLIST.KLP
LISTING.TXT	Text file produced by ERRLIST.KLP which gives locations and descriptions of errors found by PARSER and SEMAN, with corresponding lines of the program source code.

最後に, KLPMAPは, 次のファイルを生成する。

<program name>.KLP	The Pascal-SC program executable by XQP, or XQPC.
<program name>.LIN	It is used by the run-time error handler to indicate the location in the source code of the program of an instruction which generates a run-time error.

〔参考文献〕

[1] FBSoftware, Pascal-SC Documentation Release 1.1, March 1984.

[2] K.Jensen and N.Wirth, Pascal User Manual and Report, Springer Verlag (PASCAL, 原田賢一訳, 培風館, 1982) , (1975) .

[3] R.T.Gregory and D.L.Karney, A collection of Matrices for Testing Computational Algorithm, Wily-Interscience (1969) .

〔数値例〕

```
*B>XQPC FBLEIG
LEIG computes inclusions for eigenvalues and eigenvectors of real
symmetric matrices.
The commands are:

S       Submit matrix from terminal.
Z       Generate a random matrix.
Q       Quit.
W       Display approximations to eigenvalues and inclusions
        already computed.
<n>     Display the <n>-th eigenvalue/eigenvector pair (approxi-
        mations and inclusions already computed).
E <n>   Enclose the <n>-th eigenvalue/vector pair.
E  A    Enclose all eigenvalue/vector pairs.
L       Output intermediate results.
N       No intermediate results.
I       Display this information.

Here, <n> denotes an integer with value from 1 to the dimension
of the matrix.

*S
Enter dimension:
*4
Enter the left lower triangle of the symmetric matrix
by rows:
*5
*4                              [ 5  4  1  1 ]
*5                              [ 4  5  1  1 ]
*1                              [ 1  1  4  2 ]
*1                              [ 1  1  2  4 ]
*4
*1
*1
*2
*4
Give an error bound for the matrix elements:
*1.0E-12
Calculation of approximations for eigenvalues and vectors by
the Jacobi method.

The approximate eigenvalues are:
     1   9.99999999980E-01
     2   1.00000000000E+01
     3   5.00000000002E+00
     4   2.00000000000E+00
*E A
 1.  Eigenvalue:
     1   9.99999999980E-01    [   9.9999999994E-01,   1.0000000001E+00]
 1.  Eigenvector:
     1   7.07106781186E-01    [   7.0710678117E-01,   7.0710678120E-01]
     2  -7.07106781188E-01    [ -7.07106781188E-01, -7.07106781188E-01]
     3   0.00000000000E+00    [          -1.0E-11,            1.0E-11]
     4   0.00000000000E+00    [          -1.0E-11,            1.0E-11]
 2.  Eigenvalue:
     2   1.00000000000E+01    [  9.99999999997E+00,  1.00000000002E+01]
 2.  Eigenvector:
     1   6.32455532034E-01    [   6.3245553203E-01,   6.3245553205E-01]
     2   6.32455532032E-01    [   6.3245553202E-01,   6.3245553205E-01]
     3   3.16227766018E-01    [   3.1622776600E-01,   3.1622776603E-01]
     4   3.16227766017E-01    [   3.1622776600E-01,   3.1622776603E-01]
 3.  Eigenvalue:
     3   5.00000000002E+00    [   4.9999999999E+00,   5.0000000001E+00]
 3.  Eigenvector:
     1  -3.16227766018E-01    [  -3.1622776603E-01,  -3.1622776600E-01]
     2  -3.16227766017E-01    [  -3.1622776603E-01,  -3.1622776600E-01]
     3   6.32455532033E-01    [   6.3245553203E-01,   6.3245553206E-01]
     4   6.32455532032E-01    [   6.3245553201E-01,   6.3245553206E-01]
 4.  Eigenvalue:
     4   2.00000000000E+00    [   1.9999999999E+00,   2.0000000001E+00]
 4.  Eigenvector:
     1   1.17662123800E-12    [          -1.1E-11,            1.1E-11]
     2   6.66232380000E-14    [          -1.1E-11,            1.1E-11]
     3  -7.07106781186E-01    [  -7.0710678121E-01,  -7.0710678117E-01]
     4   7.07106781188E-01    [   7.07106781188E-01,  7.07106781188E-01]
*Q
KL/P-STOP
```

```
*B>XPQC FBLINV
The program LINV calculates bounds for the inverse of a matrix.
Enter one of the following letters:

S  to enter the matrix from the console.
Z  to generate a random matrix.
H  to generate a Hilbert matrix.
Q  to quit.
L  to switch on output of intermediate results.
N  to switch off output of intermediate results.
R  to output only the floating point approximation as
   the intermediate result.
I  to display this menu.

*H
Enter dimension:
*4
Give an error bound for the matrix elements:
*1.0E-12
The matrix is:

 1. column:
[  4.19999999999E+02,   4.20000000005E+02]
[  2.09999999999E+02,   2.10000000003E+02]
[  1.39999999999E+02,   1.40000000002E+02]
[  1.04999999999E+02,   1.05000000002E+02]

 2. column:
[  2.09999999999E+02,   2.10000000003E+02]
[  1.39999999999E+02,   1.40000000002E+02]
[  1.04999999999E+02,   1.05000000002E+02]
[   8.3999999999E+01,    8.4000000001E+01]

 3. column:
[  1.39999999999E+02,   1.40000000002E+02]
[  1.04999999999E+02,   1.05000000002E+02]
[   8.3999999999E+01,    8.4000000001E+01]
[  6.99999999999E+01,   7.00000000007E+01]

 4. column:
[  1.04999999999E+02,   1.05000000002E+02]
[   8.3999999999E+01,    8.4000000001E+01]
[  6.99999999999E+01,   7.00000000007E+01]
[  5.99999999999E+01,   6.00000000006E+01]


1.  Floating-point inverse:
 3.80952382044E-02-2.85714287740E-01 5.71428576172E-01-3.33333336362E-01

-2.85714287753E-01 2.85714287892E+00-6.42857147948E+00 4.00000003248E+00

 5.71428576209E-01-6.42857147955E+00 1.54285715477E+01-1.00000000759E+01

-3.33333336389E-01 4.00000003255E+00-1.00000000760E+01 6.66666671511E+00


2.  Calculation of the inverse by the new method:

The algorithm has proved that the matrix is nonsingular, and that its
inverse is guaranteed to belong to the interval matrix:
 1. column:
[       3.8095238E-02,       3.8095240E-02]
[      -2.8571430E-01,      -2.8571427E-01]
[       5.7142854E-01,       5.7142861E-01]
[      -3.3333336E-01,      -3.3333331E-01]

 2. column:
[      -2.8571430E-01,      -2.8571427E-01]
[       2.8571427E+00,       2.8571430E+00]
[      -6.4285718E+00,      -6.4285711E+00]
[       3.9999997E+00,       4.0000003E+00]

 3. column:
[       5.7142854E-01,       5.7142861E-01]
[      -6.4285718E+00,      -6.4285711E+00]
[       1.5428570E+01,       1.5428573E+01]
[      -1.0000000SE+01,      -9.99999951E+00]

 4. column:
[      -3.3333336E-01,      -3.3333331E-01]
[       3.9999997E+00,       4.0000003E+00]
[      -1.0000000SE+01,      -9.99999951E+00]
[       6.6666663E+00,       6.6666670E+00]


Enter H,S,Z or Q.  I displays the menu.
```