

# BASICによる代数方程式系の数式及び数値処理について

小鹿丈夫, 木村聡 (大阪教育大)

## 1. はじめに

$n$ 元連立非線形代数方程式

$$f(x) = 0, \quad x = (x_1, x_2, \dots, x_n)', \quad f = (f_1, f_2, \dots, f_n)' \quad (1.1)$$

を考える。一般には、(1.1)式の厳密解を求めることは困難であり、FORTRANなどの数値処理言語を用いて各種の数値解法が提案されている。しかしながら、数値処理言語においては、数式の代入或いは数式微分など、各種の数式演算を取り扱うことが困難であり、数値処理言語にも限界がある。一方、REDUCE 3[1]に見られるように、数式演算を計算機で扱える、いわゆる数式処理言語のレベルも近年向上し、次第に実用化されるに至ってはいるが、現在のところ、数式処理能力などの点で十分とは言えない。そこで、渡部、三井及び筆者等の一人は、数値処理(FORTRAN)と数式処理(REDUCE 3)を併用することにより(これをハイブリッド処理と呼ぶ)、大規模な非線形代数方程式を効率良く解くためのパッケージNAES (Nonlinear Algebraic Equations Solver)を開発し、実用段階に至っている[2, 3, 4, 5]。

他方、マイコンの著しい普及と相俟って、BASIC言語の機能も向上しつつあり、これにつれて、マイコンでもある程度の規模の非線形代数方程式を解くための簡易なパッケージの開発も強く望まれている。そこで、本稿においては、単一の言語Nga-BASICを用いてNAESの簡易版とも言うべきパッケージOKUTを開発することを試みたので、その概要について報告する。

## 2. OKUTの基本的な機能

非線形代数方程式(1.1)の数値解を求めるに当って、パッケージOKUTでは次の手順により、数式及び数値処理を行なう(Fig.2.1参照)。

- (1) 先ず、与えられた方程式と変数間の相互関係(構造, structure)をBoole行列を用いて解析する[6]。
- (2) この構造解析の結果、(i)可解性を調べ、(ii)幾つかの互いに独立した部分方程式系に分離できる場合には、方程式を組み換え、(iii)更に各部分方程式系が階層(レベル)構造に分解出来るならば、解くべき順序に部分方程式を並べ換える。
- (3) 数値解法としてはニュートン法を用いるものとし、(i)数式的にヤコビアン行列が必要な場合には、数式微分により計算し、(ii)数値微分によりヤコビアン行列を求める場合には、E-Secant法[7]により近似的に計算する。

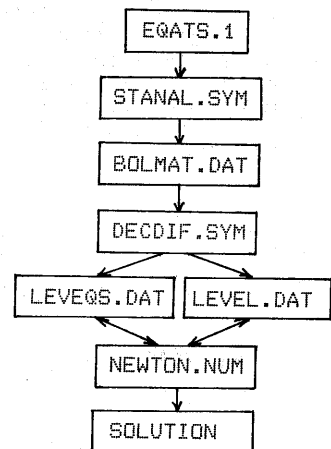


Fig.2.1. OKUTの処理手順

- (4) 対話形式によって、初期推定値を与え、逐次各レベルにおける部分方程式をニュートン法により解く。
- (5) もし重複があれば、小根、渡部、三井等が提案している deflation algorithm [8, 9] を用いて高精度に求める。

### 3. 主な文字関数

OKUT の数式処理に用いられる Nag-BASIC の主な文字関数は次のようなものである [10]。

- (i) CHR\$( <数式> ) : <数式> の値のキャラクターコードを持つ文字を与える。
- (ii) LEN( <文字列> ) : 文字列の総文字数を与える。
- (iii) RIGHT\$( X\$, N ) : 文字列 X\$ の右側から N 個の文字を取り出す。
- (iv) MID\$( X\$, M, N ) : 文字列 X\$ の先頭から M 番目の文字から後の N 個の文字を取り出す。
- (v) LEFT\$( X\$, N ) : 文字列 X\$ の左側から N 個の文字を取り出す。
- (vi) STR\$( X ) : 数式 X を文字列に変換する。
- (vii) VAL( X\$ ) : 数値を表現する文字列 X\$ を数値に変換する。
- (viii) INSTR\$( X\$, Y\$ ) : 文字列 X\$ の中から文字列 Y\$ を探し、その文字の位置を与える。

### 4. 数式及び数値処理

#### 4.1. ユーザーファイル (EQATS. 1)

いま、10元連立非線形代数方程式

```

10 REM ***** ORIGINAL EQUATIONS (EQUATS.1, ASCII SAVED) *****
20 F(1)=X(1)^2-4*X(4)+3*X(7)+1
30 F(2)=-3*X(1)+X(5)^2-1
40 F(3)=5*X(3)-X(4)^3+X(6)-2*X(8)^2
50 F(4)=X(1)^2-2*X(5)^2+7
60 F(5)=2*X(3)-3*X(8)
70 F(6)=X(7)^3-8
80 F(7)=2*X(3)-6*X(6)^4
90 F(8)=X(10)^2-36
100 F(9)=X(2)+X(8)^2
110 F(10)=6*X(9)^2+X(10)^3
120 END

```

(4.1)

を用いて、OKUT における各種の数式及び数値処理の概要について述べる。なお (4.1) 式は、ファイル EQATS. 1 にアスキー形式でセーブされているものとす。

#### 4.2. 構造解析 (STANAL. SYM) [6]

(1) 先ず (4.1) から

$$e_{ij} = \begin{cases} 1, & \text{もし変数 } x_j \text{ が方程式 } f_i \text{ に陽に含まれている場合,} \\ 0, & \text{含まれていない場合,} \end{cases}$$

とする Boolean 行列 E (方程式行列) を作る。そのためのプログラムを Fig. 4.1 に、その結果を (4.2) に示す。

```

1 REM ***** STRUCTURE ANALYSIS (STANAL.SYM) *****
3 DEFINT A-Z
4 CLEAR 2000
5 DIM F$(20),X$(20),E(20,20),V(20,20),OF(20)
6 PRINT "INPUT THE DRIVE NO. AND FILE NAME OF THE EQUATIONS (ASCII SAVED) !"
7 INPUT EQUATIONS$
10 OPEN EQUATIONS$ FOR INPUT AS #1
20 LINE INPUT #1,A$
25 C$=""
30 B$=A$
35 D$="F("
40 IF LEFT$(B$,LEN(D$))<>D$ GOTO 120
45 B$=RIGHT$(B$,LEN(B$)-LEN(D$))
50 IF LEFT$(B$,1)=")" GOTO 90
55 IF ASC(LEFT$(B$,1))<&H30 OR ASC(LEFT$(B$,1))>&H39 GOTO 150
60 C$=C$+LEFT$(B$,1)
70 B$=RIGHT$(B$,LEN(B$)-1)
80 GOTO 50
90 IF C$="" GOTO 150
95 IF CMAX<VAL(C$) THEN CMAX=VAL(C$)
100 F$(VAL(C$))=RIGHT$(B$,LEN(B$)-2)
105 PRINT
110 PRINT "f(";VAL(C$);")=";F$(VAL(C$));GOTO 150
120 IF LEN(B$)=0 GOTO 150
130 B$=RIGHT$(B$,LEN(B$)-1)
140 GOTO 40
150 IF EOF(1)=0 GOTO 20
160 CLOSE
165 REM ***** GENERATION OF THE EQUATION MATRIX E *****
170 NX=CMAX
175 CMAX=0
180 FOR I=1 TO NX
182 X$(I)="X("+RIGHT$(STR$(I),LEN(STR$(I))-1)+")"
184 NEXT I
240 PRINT
320 FOR I=1 TO NX
340 FOR J=1 TO NX
350 IF INSTR(F$(I),X$(J))<>0 THEN E(I,J)=1
360 PRINT E(I,J);
390 NEXT J
400 PRINT
410 NEXT I

```

Fig. 4.1

$$E = \begin{matrix} & \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} \end{matrix} \\ \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{matrix} \quad (4.2)$$

(2) 次に互いに独立した部分方程式系(サブシステム)があるかないかを調べる[11]. 方程式行列Eの第j列において"1"がある行同志の論理和を求め, 各行列には"1"が1個存在するようにするまで, この操作を繰返す. その結果を(4.3)に示す.

$$E^* = \begin{matrix} & & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} \\ \begin{matrix} \text{7} \\ (\cup f_i) \\ i=1 \end{matrix} \cup f_9 & \left[ \begin{array}{cccccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array} \right] \\ f_8 \cup f_{10} & \end{matrix} \quad (4.3)$$

これから, 原方程式(4.1)は, 互いに独立した方程式  $f_1 \sim f_7, f_9$  と変数  $x_1 \sim x_8$  からなる第1サブシステムと  $f_8, f_{10}$  と  $x_9, x_{10}$  からなる第2サブシステムから構成されていることが分かる.

(3) 独立したサブシステムごとに(4.2)を並べ換えると, (4.4)が得られる. ここで, どの方程式からどの変数(output variable)が見掛け上解き得るかを定める.

$$\hat{E} = \begin{matrix} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & rS \\ \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \\ f_{10} \\ cS \end{matrix} & \left[ \begin{array}{cccccccccc|cc} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 3 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 4 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 3 & 1 & 3 & 2 & 2 & 2 & 2 & 3 & 1 & 2 & 0 \end{array} \right] \end{matrix} \quad (4.4)$$

(4.4)において, 最小のrS(row sum, 行方向の"1"の数)は1で, 第6行と第9行である. このことは,  $f_6$  及び  $f_9$  には  $x_7$  及び  $x_{10}$  のみが各々陽に含まれていることを示して, これらの変数が各々のoutput variableである. 第6, 9行及び第7, 10列を(4.4)から消去する. 最小のcS(column sum)を持つ列についても同様の操作を行う. 最終的に得られた結果を(4.5)に示す.

$$\begin{matrix} f_1 - x_4, f_2 - x_1, f_3 - x_3, f_4 - x_5, f_5 - x_8, f_6 - x_7, f_7 - x_6, \\ f_8 - x_{10}, f_9 - x_2, f_{10} - x_9. \end{matrix} \quad (4.5)$$

(4) (4.5)の結果を用いて, 変数  $x_i$  に対応する  $f_j$  に書き換えた上で, (4.4)の第1列における方程式の順序に並べ換える. 第1サブシステムの結果を(4.6)に示す.

$$E^{(1)} = \begin{matrix} & f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_9 \\ \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_9 \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \end{matrix} \quad (4.6)$$

(5)  $n \times n$  の Boole 行列  $E$  の到達行列  $R$  は

$$R = E^{n-1} \quad (4.7)$$

で定義され、この  $R_{ij}$  要素が "1" であることは、 $(n-1)$  個以内の経路を通して  $f_j$  から  $f_i$  へ到達できることを示す。

第1サブシステムの到達行列  $R^{(1)}$  は

$$R_1^{(1)} = \begin{matrix} & f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_9 & rs(f_i) & ds(f_i) & level \\ \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_9 \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} & \begin{matrix} 4 \\ 2 \\ 7 \\ 2 \\ 7 \\ 1 \\ 7 \\ 8 \end{matrix} & \begin{matrix} 1 \\ 5 \\ -3 \\ 5 \\ -3 \\ 5 \\ -3 \\ -7 \end{matrix} & \begin{matrix} \\ 1-1 \\ \\ 1-1 \\ \\ 1-2 \\ \\ \end{matrix} \end{matrix} \quad (4.8)$$

$$cs(f_i) = \begin{matrix} 5 & 7 & 4 & 7 & 4 & 6 & 4 & 1 \end{matrix}$$

となる。

(6) いま到達行列  $R$  において、 $rs$  (特に入度数と呼ぶ) と  $cs$  (出度数) との差を

$$ds(f_i) = cs(f_i) - rs(f_i), \quad i=1, 2, \dots, n. \quad (4.9)$$

とおけば、 $ds$  の値が大きいかほど  $f_i$  は上位レベルに位置することがわかる。

(4.8) 式の第2, 4, 6 行の  $ds$  は最大値 5 であり、第1レベルとす。更に、第2行と第4行は全く同じ行ベクトルであり、第6行のそれとは相異なる。したがって、 $f_2$  と  $f_4$  を第1レベルの第1グループとし、 $f_6$  を第2グループとする。次に、第2, 4, 6 行及び列を各々消去し、同様の手順を繰返す。最終的に得られた結果を(4.10)に示す。

| サブシステム | レベル  | グループ | 方程式番号 |    |   |   |   |   |   |   |
|--------|------|------|-------|----|---|---|---|---|---|---|
| E=     | 1    | 1    | 1     | 2  | 4 | 0 | 0 | 0 | 0 |   |
|        | 1    | 1    | 2     | 6  | 0 | 0 | 0 | 0 | 0 |   |
|        | 1    | 2    | 1     | 1  | 0 | 0 | 0 | 0 | 0 |   |
|        | 1    | 3    | 1     | 3  | 5 | 7 | 0 | 0 | 0 |   |
|        | 1    | 4    | 1     | 9  | 0 | 0 | 0 | 0 | 0 |   |
|        | 2    | 1    | 1     | 8  | 0 | 0 | 0 | 0 | 0 |   |
|        | 2    | 2    | 1     | 10 | 0 | 0 | 0 | 0 | 0 |   |
|        | 変数番号 |      |       |    |   |   |   |   |   |   |
|        | 1    | 1    | 1     | 1  | 5 | 0 | 0 | 0 | 0 | 0 |
|        | 1    | 1    | 2     | 7  | 0 | 0 | 0 | 0 | 0 | 0 |
| 1      | 2    | 1    | 4     | 0  | 0 | 0 | 0 | 0 | 0 |   |
| 1      | 3    | 1    | 3     | 8  | 6 | 0 | 0 | 0 | 0 |   |
| 1      | 4    | 1    | 2     | 0  | 0 | 0 | 0 | 0 | 0 |   |
| 2      | 1    | 1    | 10    | 0  | 0 | 0 | 0 | 0 | 0 |   |
| 2      | 2    | 1    | 9     | 0  | 0 | 0 | 0 | 0 | 0 |   |

(4.10)

なお OKUT においては, (4.10)の結果は BOLMAT.DAT にセーブされる。

### 4.3. 方程式の並び換え (DECDIF.SYM)

4.2 における構造解析に従って, (i) (4.1)を解くべき順序に並び換え, (ii) 計算を実行するために必要な各種の命令を与え, (iii) Jacobean 行列を計算し, (iv) E-Secant 法で計算するための計算式を生成する。その結果を LEVEQS.DAT にセーブする。方程式を並び換えるためのプログラムの一部を Fig. 4.2 に, また LEVEQS.DAT に生成された  $f_2, f_4$  および  $x_1, x_5$  からなる第 1 サブシステム, 第 1 レベル, 第 1 グループを解くためのプログラムを Fig. 4.3 に示す。

```

5500 FOR J=1 TO LT
5510 XX$(J)=X$(OF(E(IST,J+3)))
5520 XC$(J)="X"+FNL$(STR$(J))
5530 NEXT J
5550 FOR K=1 TO LT
5560 FC$(K)=F$(E(IST,K+3))
5570 FOR J=1 TO LT
5580 INS=INSTR(FC$(K),XX$(J))
5590 IF INS=0 GOTO 5620
5600 FC$(K)=LEFT$(FC$(K),INS-1)+XC$(J)+RIGHT$(FC$(K),LEN(FC$(K))-LEN(XX$(J))-INS+1)
5610 GOTO 5580
5620 NEXT J
5630 NEXT K
5700 FOR I=1 TO LT
5710 LIN$=FNL$(STR$(LC))+ " DEF FNF"+FNL$(STR$(I))+ "("
5720 FOR J=1 TO LT
5730 IF J>1 THEN LIN$=LIN$+", "
5740 LIN$=LIN$+XC$(J)
5750 NEXT J
5760 LIN$=LIN$+"")="+FC$(I)+CR$
5770 PRINT #1,LIN$:LC=LC+10
5780 NEXT I

```

Fig. 4.2

```

990 ON IST GOSUB 1000,2000,3000,4000,5000,6000,7000:RETURN
1000 ON JSF GOSUB 1010,1070,1200,1230,1260:RETURN
1010 DEF FNF1(X1,X2)=-3*X1+X2^2-1
1020 DEF FNF2(X1,X2)=X1^2-2*X2^2+7
1030 PRINT #1," SOLVE f2 f4 about x1 x5"
1040 INPUT "symbolic dif yes=1 / no=0";JF
1050 INPUT " X(1) X(5)";XL(1),XL(2)
1053 PRINT #1,"initial guess ";:PRINT #1," X(1)=";:PRINT #1,USING "+#.###^";XL(1),:PRINT #1," X(5)=";:PRINT #1,USING "+#.###^";XL(2),
1056 PRINT #1,
1060 RETURN
1070 IF JF<>1 GOTO 1130
1080 A(1,1)=-3
1090 A(1,2)=2*XL(2)
1100 A(2,1)=2*XL(1)
1110 A(2,2)=-2*2*XL(2)
1120 GOTO 1200
1130 IF XL(1)>1 THEN DD=DE*XL(1) ELSE DD=DE
1140 A(1,1)=(FNF1(XL(1)+DD,XL(2))-FNF1(XL(1),XL(2)))/DD
1150 A(2,1)=(FNF2(XL(1)+DD,XL(2))-FNF2(XL(1),XL(2)))/DD
1160 IF XL(2)>1 THEN DD=DE*XL(2) ELSE DD=DE
1170 A(1,2)=(FNF1(XL(1),XL(2)+DD)-FNF1(XL(1),XL(2)))/DD
1180 A(2,2)=(FNF2(XL(1),XL(2)+DD)-FNF2(XL(1),XL(2)))/DD
1200 F(1)=FNF1(XL(1),XL(2))
1210 F(2)=FNF2(XL(1),XL(2))
1220 RETURN
1230 PRINT #1,"in loop ";:PRINT #1," X(1)=";:PRINT #1,USING "+#.###^";XL(1),:PRINT #1," X(5)=";:PRINT #1,USING "+#.###^";XL(2),
1240 PRINT #1,
1250 RETURN
1260 PRINT #1,:PRINT #1," ANS"
1270 PRINT #1,"X(1)=";:PRINT #1,USING "+#.#####^";XL(1):X(1)=XL(1)
1280 PRINT #1,"X(5)=";:PRINT #1,USING "+#.#####^";XL(2):X(5)=XL(2)
1290 PRINT #1,
1300 RETURN

```

Fig 4.3

#### 4.4. 数式微分

微分における基本的な公式としては、

$$\frac{d}{dx}(g(x)h(x)) = g'(x)h(x) + g(x)h'(x) \quad (4.11a)$$

$$\frac{d}{dx} g(h(x)) = g'(x)h'(x) \quad (4.11b)$$

がある。OKUTにおいては、3.4 で述べた文字関数を用いて与えられた数式を文字式に変換して微分処理を行なっている。先ず、例題を示す。

Ex.1      8003 F#="X(1)^3+X(2)+1":X#="X(1)"  
Ok  
run  
FD= 3\*X(1)^(3-1) (4.12)

Ex.2      8002 X#="X(1)":F#="(X(1)^3+X(2)+1)\*(X(3)+2)"  
Ok  
run  
FD= (X(3)+2)\*(3\*X(1)^(3-1)) (4.13)

8003 F#="(X(1)^3+X(2)+1)^4+3\*(X(3)+2)":X#="X(1)

Ex. 3

Ok

run

FD= (X(3)+2)\*(4\*(X(1)^3+X(2)+1)^(4-1)\*(3\*X(1)^(3-1)))

(4.14)

いま、(4.13)の例題における処理手順をFig.4.4に、Fig.4.4における処理4.の部分に対応するプログラムをFig.4.5に、また微分した結果(FD)を生成するためのプログラムをFig.4.6にそれぞれ示す。

run

1. C1=( F2=(X(1)^3+X(2)+1)\*(X(3)+2) KE= C2= LK= 1 FD=

1. C1=(X F2=X(1)^3+X(2)+1)\*(X(3)+2) KE= C2= LK= 1 FD=

1a. C1=(X F2=^3+X(2)+1)\*(X(3)+2) KE= C2= LK= 1 FD=

2. C1= F2=^3+X(2)+1)\*(X(3)+2) RK= 0 KE=X(1)

3. C1= F2=^3+X(2)+1)\*(X(3)+2) RK= 0 KE=(X(1)

4. C1= F2=3+X(2)+1)\*(X(3)+2) LK= 1 KE=(X(1)^ C2= LK= 1 FD=

4. C1= F2=+X(2)+1)\*(X(3)+2) LK= 1 KE=(X(1)^3 C2= LK= 1 FD=

4. C1= F2=X(2)+1)\*(X(3)+2) LK= 1 KE=(X(1)^3+ C2= LK= 1 FD=

4. C1= F2=(2)+1)\*(X(3)+2) LK= 1 KE=(X(1)^3+X C2= LK= 1 FD=

4. C1= F2=2)+1)\*(X(3)+2) LK= 2 KE=(X(1)^3+X( C2= LK= 2 FD=

4. C1= F2=)+1)\*(X(3)+2) LK= 2 KE=(X(1)^3+X(2 C2= LK= 2 FD=

4. C1= F2=+1)\*(X(3)+2) LK= 1 KE=(X(1)^3+X(2) C2= LK= 1 FD=

4. C1= F2=1)\*(X(3)+2) LK= 1 KE=(X(1)^3+X(2)+ C2= LK= 1 FD=

4. C1= F2=)\*X(3)+2) LK= 1 KE=(X(1)^3+X(2)+1 C2= LK= 1 FD=

4. C1= F2=)\*X(3)+2) LK= 0 KE=(X(1)^3+X(2)+1) C2= LK= 0 FD=

7. C1= F2=(X(3)+2) KE=(X(1)^3+X(2)+1) C2=\* LK= 0 FD= PO=

7. C1= F2=X(3)+2) KE=(X(1)^3+X(2)+1) C2=\*( LK= 1 FD= PO=

7. C1= F2=(3)+2) KE=(X(1)^3+X(2)+1) C2=\*(X LK= 1 FD= PO=

7. C1= F2=3)+2) KE=(X(1)^3+X(2)+1) C2=\*(X( LK= 2 FD= PO=

7. C1= F2=)+2) KE=(X(1)^3+X(2)+1) C2=\*(X(3 LK= 2 FD= PO=

7. C1= F2=+2) KE=(X(1)^3+X(2)+1) C2=\*(X(3) LK= 1 FD= PO=

7. C1= F2=2) KE=(X(1)^3+X(2)+1) C2=\*(X(3)+ LK= 1 FD= PO=

7. C1= F2=) KE=(X(1)^3+X(2)+1) C2=\*(X(3)+2 LK= 1 FD= PO=

7. C1= F2=) KE=(X(1)^3+X(2)+1) C2=\*(X(3)+2 LK= 1 FD= PO=

7. C1= F2=) KE=(X(1)^3+X(2)+1) C2=\*(X(3)+2 LK= 0 FD= PO=

9. NES= 1 PO= 0 F2=X(1)^3+X(2)+1 C2=\*(X(3)+2) FD=

10. C1= F2=X(1)^3+X(2)+1 KE=(X(1)^3+X(2)+1) C2=(X(3)+2) LK= 0 FD=(X(3)+2)\*

( PD=

1. C1=X F2=X(1)^3+X(2)+1 KE=(X(1)^3+X(2)+1) C2= LK= 0 FD=(X(3)+2)\*

1a. C1=X F2=^3+X(2)+1 KE=(X(1)^3+X(2)+1) C2= LK= 0 FD=(X(3)+2)\*



```

6. PO= 3  F2= +X(2)+1  KE=X(1)
11. C1= F2=+X(2)+1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))  PO=3
1. C1=+ F2=+X(2)+1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X F2=X(2)+1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X( F2=(2)+1  KE=X(1)  C2= LK= 1  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X(2 F2=2)+1  KE=X(1)  C2= LK= 1  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X(2) F2=)+1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X(2)+ F2=+1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))
1. C1=+X(2)+1 F2=1  KE=X(1)  C2= LK= 0  FD=(X(3)+2)*(3*X(1)^(3-1))
13. FD= (X(3)+2)*(3*X(1)^(3-1))

```

Fig. 4.4

```

8190 RK=LK
8200 R$=RIGHT$(C1$,1)
8210 IF R$="" THEN RK=RK+1
8220 IF R$="(" THEN RK=RK-1
8230 C1$=LEFT$(C1$,LEN(C1$)-1)
8235 PRINT
8239 PRINT"2. C1=";C1$;" F2=";F2$;" RK=";RK;" KE=";KE$
8240 KE$=R$+KE$
8247 PRINT"3. C1=";C1$;" F2=";F2$;" RK=";RK;" KE=";KE$
8250 IF RK<>0 GOTO 8200
8260 REM f-check exp-p
8270 L$=LEFT$(F2$,1)
8280 IF L$="(" THEN LK=LK+1
8290 IF L$=")" THEN LK=LK-1
8300 KE$=KE$+L$
8310 F2$=RIGHT$(F2$,LEN(F2$)-1)
8315 PRINT "4. C1=";C1$;" F2=";F2$;" LK=";LK;" KE=";KE$;" C2=";C2$;" LK
";LK;" FD=";FD$

```

Fig. 4.5

```

9210 IF C2$="" THEN FD$=FD$+PO$+"*"+KE$+"^("+"PO$+"-1)";GOTO 9230
9220 FD$=FD$+C2$+"*"+PO$+"*"+KE$+"^("+"PO$+"-1)"
9230 PRINT"11. C1=";C1$;" F2=";F2$;" KE=";KE$;" C2=";C2$;" LK=";LK;" FD=";F
D$;" PO=";PO$

```

Fig. 4.6

#### 4.5. 数値処理 (NEWTON. NUM)

OKUTでは、与えられた非線形方程式系をニュートン法

$$\left(\frac{\partial f^{(k)}(x)}{\partial x}\right)(x^{(k+1)} - x^{(k)}) = -f^{(k)}(x), \quad k=0, 1, 2, \dots, \quad (4.15)$$

により、繰返し解く。この場合、ヤコビアン行列 $(\partial f/\partial x)$ を、4.4で述べた数値微分により数式的に求める場合と数値微分により求める場合とがあり、いずれを用いるかは計算実行時に選択出来るようになっている。なお、線形方程式(4.15)はGAUSSの消去法により解いている。また、収束判定条件は、

$${}^{k+1}G = \left[ \frac{1}{n} f'(x^{(k+1)}) f(x^{(k+1)}) \right]^{1/2} \leq \varepsilon \quad (4.16)$$

で行なっている。

#### 5. 数値例

ここでは、(4.1)を数式によるJacobian行列を用いて、倍精度で解く。その結果の一部を(5.1)に示す。なお、各グループにおける収束判定値は $\varepsilon=10^{-14}$ であり、

EPS は (4.16) の  $\epsilon$  を表わす。

```
SOLVE f2 f4 about x1 x5
initial guess          X(1)=+2.000D+00 X(5)=+3.000D+00
EPS=+8.327D-01
in loop               X(1)=+5.000D-01 X(5)=+1.917D+00
EPS=+1.402D-01
in loop               X(1)=+9.500D-01 X(5)=+1.963D+00
EPS=+1.006D-03
in loop               X(1)=+9.994D-01 X(5)=+2.000D+00
EPS=+2.455D-07
in loop               X(1)=+10.000D-01 X(5)=+2.000D+00
EPS=+0.000D+00
```

```
ANS
X(1)=+9.99999999999998D-01
X(5)=+2.00000000000000D+00 (5.1)
```

```
SOLVE f6 about x7
initial guess          X(7)=+3.000D+00
EPS=+4.108D+00
in loop               X(7)=+2.296D+00
EPS=+4.471D-01
in loop               X(7)=+2.037D+00
EPS=+7.843D-03
in loop               X(7)=+2.001D+00
EPS=+2.560D-06
in loop               X(7)=+2.000D+00
EPS=+2.736D-13
in loop               X(7)=+2.000D+00
EPS=+0.000D+00
```

```
ANS
X(7)=+2.00000000000000D+00
```

## 6. おわりに

本稿においては、 $N_{gg}$ -BASIC を用いた非線形代数方程式解法用パッケージ OKUT の開発を行なったので、その概要について述べた。この OKUT は、数式処理と数値処理とを併用することにより、効率良く解くことを意図したものである。今後の課題としては、(i) 重根を解くための deflation algorithm を組み込む (既に一部は組み込まれている)、(ii) 各種のサブルーチンを機械語にすることにより処理のスピード化を図ることである。

## 参考文献

1. A.C.Hearn, REDUCE 3 Users Manual, The Rand Corporation, Santa Monica, C.A., 1983.
2. T.Ojika, S.Watanabe and T.Mitsui, NAES Users Manual, (in preparation).
3. 小底, 渡部, 三井, 連立非線形代数方程式解法用パッケージ NAES とその手法, 数理科学, No.218, 1981.
4. 小底, 三井, 渡部, 非線形代数方程式系の解法パッケージ NAES, 電子通信学会資料 CAS 82-122, 1982.
5. S.Watanabe, Hybrid Manipulations for the Solution of Systems of Nonlinear Algebraic Equations, Publ. RIMS Kyoto Univ. (in print).
6. T.Ojika, Structure Analysis for Large Scale Nonlinear Equations, Memo. Osaka Kyoiku Univ. Vol.32, 1983.
7. S.Watanabe, T.Ojika and T.Mitsui, On the Quadratic Convergence Properties of the  $\epsilon$ -Secant Method for the Solution of Systems of Nonlinear Equations and its Application to a Chemical Reaction Problem, JMAA, Vol.94, 69-84, 1983.
8. T.Ojika, Deflation Algorithm for the Multiple Roots of Simultaneous Equations, Memo. Osaka Kyoiku Univ., Vol.30, 197-209, 1982.
9. T.Ojika, S.Watanabe and T.Mitsui, Deflation Algorithm for the Roots of a System of Nonlinear Equations, JMAA, Vol.96, 1983.
10. PC-8801 BASIC リファレンス マニュアル, NEC
11. W.P.Ledet and D.M.Himmelbrau, Decomposition Procedures for the Solving of Large Scale Systems, Advances in Chem. Eng., 8, 185-254, 1970