CP/M FORTRAN -80程式語言

下冊

第八章 FORTRAN-80 在數值分析上的應用

FORTRAN 語言，爲研究數值分析不可缺少的工具，在這一章中，我們將研究數值分析的各種技巧，如陣列之應用、微分方程式之解法等，並借助於FORTRAN-80，而直接在APPLE II 上執行。

§8.1　非線性方程式解法

在這裏，所要討論的是解方程式的根。

§8.1.1　半區間尋根法

我們所要研究的第一個方法爲半區間尋根法。

今以下列方程式來加以討論：

$$f(x) = x^3 + x^2 - 3x - 3 = 0$$

$\because f(1) = -4 < 0$，$f(2) = 3 > 0$，又 $f(x)$ 爲連續函數，所以至少有一根在（1，2）區間內。

第八章
FORTRAN-80
在數值分析上的應用

FORTRAN 語言，爲研究數值分析不可缺少的工具，在這一章中，我們將研究數值分析的各種技巧，如陣列之應用、微分方程式之解法等，並借助於FORTRAN-80，而直接在APPLE II上執行。

§8.1　非線性方程式解法

在這裏，所要討論的是解方程式的根。

§8.1.1　半區間尋根法

我們所要研究的第一個方法爲半區間尋根法。

今以下列方程式來加以討論：

$$f(x)=x^3+x^2-3x-3=0$$

$\because f(1)=-4<0$，$f(2)=3>0$，又$f_{(x)}$爲連續函數，所以至少有一根在（1，2）區間內。

先求 f（1.5）＝−1.875，再與 f(1)，f(2) 比較，因 f（1.5）與 f(2) 異號，所以根在（1.75，2） 區間，如此繼續下去，可得一非常小的區間，而根必在此區間內，此例，正確的根為：

$$x = \sqrt{3} = 1.73205 \cdots$$

而由表1　知所求近似值，若重複次數愈多，則愈接近正確值。

如果，我們要求所找之根 x_i，其值 $|f(x_i)| < \in$，$\in$ 為任意選定的微小正數，那麼，我們必須繼續尋根，直到滿足 $|f(x_i)| < \in$ 為止。

表 8-1　用半區間尋根法解 $f(X)=x^3+x^2-3x-3=0$

重複次數	x_1	x_2	x_3	$f(x_1)$	$f(x_2)$	$f(x_3)$	X_3 之最大誤差
1	1	2	1.5	−4.0	3.0	−1.875	0.5
2	1.5	2	1.75	−1.875	3.0	0.17187	0.25
3	1.5	1.75	1.625	−1.875	0.17187 . . .	−0.94335 . . .	0.125
4	1,625	1.75	1.6875	−0.94335 . . .	0.17187	0.40942	0.0625
5	1.6875	1.75	1.71875	−0.40942 . . .	0.17187 . . .	−0.12478	0.03125
6	1.71875	1.75	1.73437 . . .	−0.12478 . . .	0.17187 . . .	−0.0219	0.01562
7	1.71875	1.73437 . . .	1.72656 . . .				0.00781
	.	.	.				
	.	.	.				
	.	.	.				
∞			1.73205 . . .			−0.00000 . . .	

* 在 5 次重複後 X_3 之實際誤差為 0.01330

§8.1.2 假位法

若要加快求根之速率，可使用假位法。

假設 f(x) 在 (x_1 , x_2) 區間為線性，且 f (x_1) 與 f (x_2) 異號，則由圖 8-1 中知：

$$\frac{x_2 - x_3}{x_2 - x_1} = \frac{f(x_2)}{f(x_2) - f(x_1)}$$

$$\therefore \quad x_3 = x_2 - \frac{f(x_2)}{f(x_2) - f(x_1)} (x_2 - x_1) \qquad (1)$$

然後算出 f (x_3)，並將 f (x_3) 之符號與 f (x_1) 、 f (x_2)之符號相比，若與 f (x_1) 符號相異，則根在 (x_1 , x_3) 區間，若與 f (x_2) 相異，則在 (x_3 , x_2) 區間，知道區間後，再重複此方法，直到滿足：

$|f(x_i)| < \in$ ，其中$\in$為微小正數，如 10^{-5}

表 8-2與 表 8-1 比較知 ，以假位法處理的結果，顯然假位法趨近得更快。

表 8-2 用假位法解 $f(X) = x^3 + x^2 - 3x - 3 = 0$

重複次數	x_1	x_2	x_3	$f(x_1)$	$f(x_2)$	$f(x_1)$
1	1.0	2.0	1.57142	−4.0	3.0	−1.36449
2	1.57142	2.0	1.70540	−1.36449	3.0	−0.24784
3	1.70540	2.0	1.72788	−0.24784	3.0	−0.03936
4	1.72788	2.0	1.73140	−0.03936	3.0	−0.00615
5	1.73140	2.0	1.73194*			

*在 5 次重複後 X_3 的誤差爲 0.00011

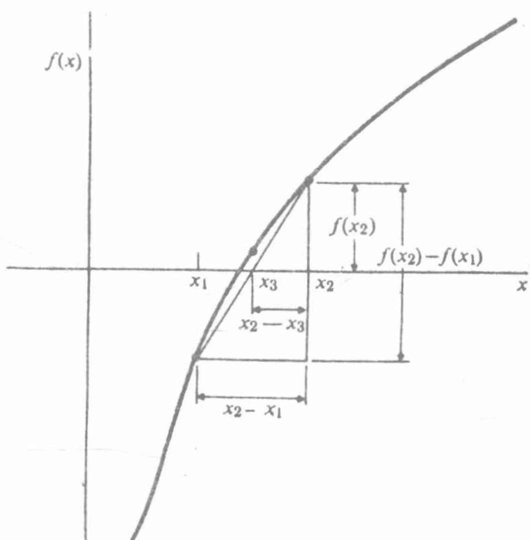

圖 8-1 假位法之圖示

§8.1.3 牛頓法

牛頓法爲解方程式使用最廣泛的一種方法。

今以圖 8-2 加以說明。

x_1 爲預估值，由 $(x_1, f(x_1))$ 點在 f 上繪一切線，交 x 軸於 x_2 ，x_2 即爲下一近似值，顯然：

$$\tan\theta = f'(x_1) = \frac{f(x_1)}{x_1 - x_2}$$

$$\therefore x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \qquad (2)$$

所以牛頓法之公式爲：

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad , n = 1, 2, 3, \cdots \qquad (3)$$

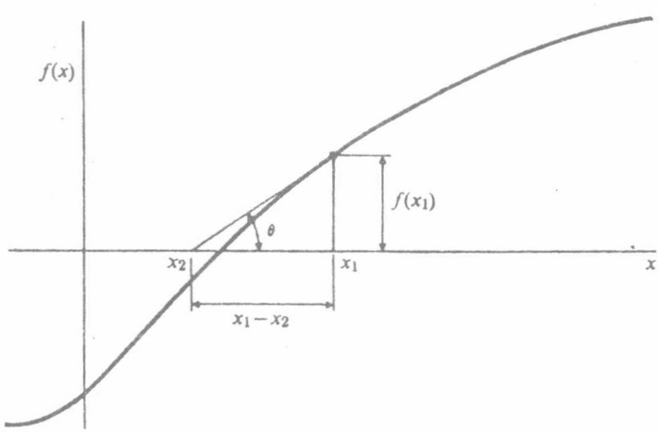

圖 8-2　牛頓法之圖示

例：試用牛頓法解 $x^2 - c = 0$ 之根，以求得 c 之平方根。

設　$f(X) = x^2 - o$

　　$f'(X) = 2x$

所以牛頓疊代式爲：

$$x_{n+1} = x_n - \frac{x_n^2 - c}{2x_n}$$

或整理成：

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{c}{x_n}\right)$$

假如要算 $\sqrt{24}$ ，我們可設定其初值＝ 1，則 $x_1 = 1$

$$x_2 = \frac{1}{2}(1 + 24) = 12.5$$

$$x_3 = \frac{1}{2}\left(12.5 + \frac{24}{12.5}\right) = 7.21$$

$$x_4 = \frac{1}{2}\left(7.21 + \frac{24}{7.21}\right) = 5.2693$$

$$x_5 = \frac{1}{2}\left(5.2693 + \frac{24}{5.2693}\right) = 4.9119$$

$$x_6 = \frac{1}{2}\left(4.9119 + \frac{24}{4.9119}\right) = 4.8989$$

$\vdots$

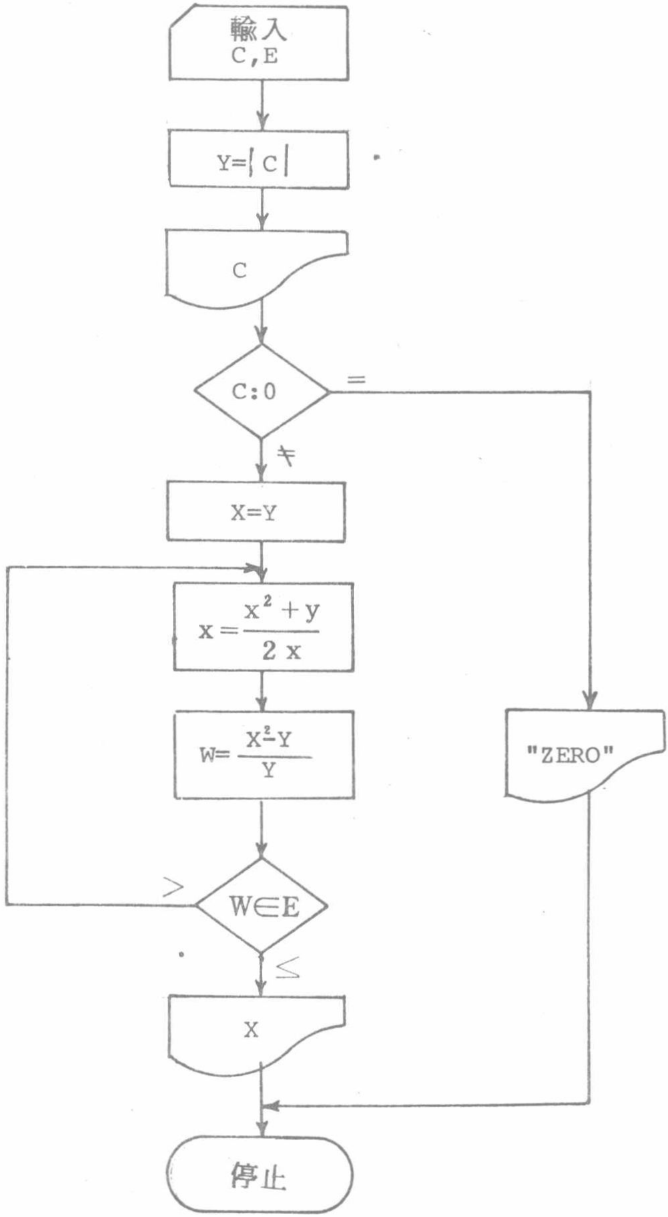

圖 8-3　以牛頓法求平方根之流程圖

圖 8-3 爲利用此法解 $x^2 - c = 0$ 之流程圖，注意此疊代步驟一直進行，直到滿足下式：

$$\left| \frac{x^2 - c}{c} \right| \leq \in$$

∈爲一微小正數。

利用此法所設計出之FORTRAN 程式，及該程式在 APPLE II 上編譯，載入連結並進而執行之過程，均示於下：

第一次執行時，輸入 c = 24，∈= 0.00001，而求出之 x = 4.89900。

第二次執行時，輸入 c = 65，∈= 0.00001，而求出之 x = 8.06226。

```
A>
A>TYPE NEWTON.FOR
C       NEWTON-RAPHSON METHOD FOR FINDING SQUARE ROOT
        READ(1,2)C,E
2       FORMAT(2F8.0)
        Y=ABS(C)
10      WRITE(2,7)C
7       FORMAT(10X,4H C= ,F12.5)
        IF (C) 11,12,11
12      WRITE(2,17)
17      FORMAT(8H X= ZERO//)
        GOTO40
11      X=Y
14      X=(X*X+Y)/(2.*X)
        W=ABS((X*X-Y)/Y)
        IF(W-E)30,30,14
30      WRITE(2,50)X
50      FORMAT(10X,4H X= ,F12.5//)
40      STOP
        END
```

```
A>F80 N,=NEWTON
$MAIN
L80

Link-80  Vers. 3.41
Copyright (C) 1980 by Microsoft
Created: 28-Dec-80

*N,N/N
Data    0103    0257    <  340>

 FORLIB RQUEST
-$AB    01E8    -$DB    0206
-$I1    023D    -$INIT  0163
-$L1    01DC    -$MB    01FA
-$ND    0240    -$R2    016C
-$SB    021E    -$ST    0243
-$T1    0224    -$W2    0232
-ABS    0212
   13 Undefined Global(s)
29314 Bytes Free

*FORLIB/S/E

Data    0103    1DFD    < 7418>

 FORLIB RQUEST
20856 Bytes Free
[015F   1DFD       29]

A>
A>N
      240.00001

          C=      24.00000
          X=       4.89900

 STOP

A>N
65.     0.00001
```

```
          C=        65.00000
          X=         8.06226

  STOP

 A>
```

例：試用半區間尋根法，求下列方程式在（0，1）區間內的根：

$$f(X) = e^{-x} - \sin(\pi x/2) = 0$$

圖8-4爲使用半區間尋根法之流程圖，圖中A、B兩變數爲區間之左右二極限值，∈爲誤差容許度，亦即，直到條件 $|f(x_i)| \leq \in$ 被滿足爲止，否則，繼續尋根之重複步驟。

流程圖中 f(A)・f(X)：0 是用來比較 f(A)與 f(X) 之符號，若 f(A)・f(X) 爲正，則 f(A)與 f(X) 同號，反之，若爲負，則 f(A) 與 f(X) 異號，若 f(A)・f(X)＝0 。則 f(A) 必等於 0 ，故A 爲一眞正的根。

所要求印成之格式爲：

(I)	X(I)	SIN	EXP	F(X)
———	———	———	———	———
———	———	———	———	———
———	———	———	———	———
(I5)	(F15.8)	(F15.8)	(F15.8)	(F20.8)

依圖8-4之流程圖，所設計之FORTRAN 程式，及其在APPLE Ⅱ 上發展之結果如下：

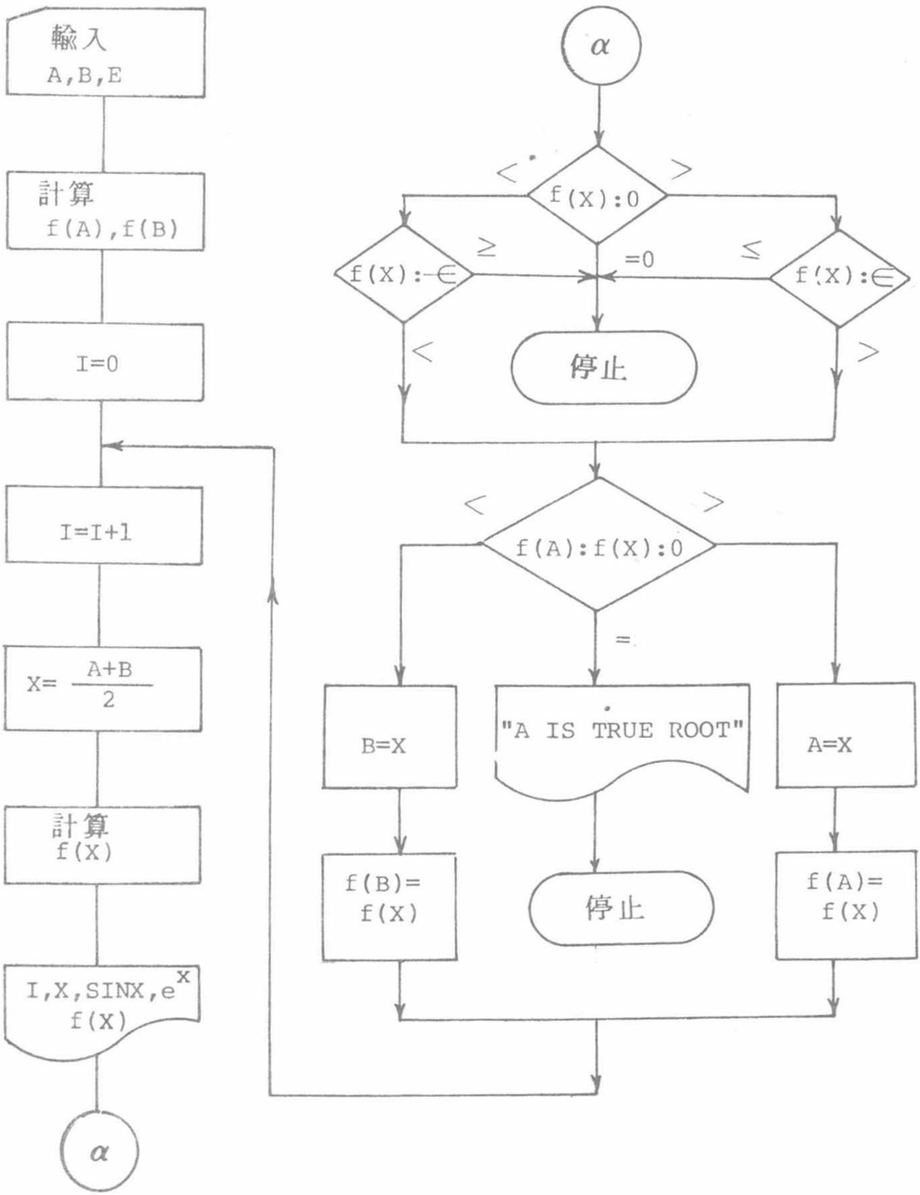

圖 8-4 半區間尋根法之流程圖

```
A>
A>TYPE HALF.FOR
C
C  HALF INTERVAL RESEARCH
C
   READ(1,22)A,B,EPS
22 FORMAT(3F10.4)
   PI=3.14159265
   FA=EXP(-A)-SIN(PI*A/2.)
   FB=EXP(-B)-SIN(PI*B/2.)
   WRITE(2,81)
81 FORMAT(5X,1HI,9X,4HX(I),11X,3HSIN,12X,3HEXP,17X,4HF(X)/
   I=0
7  I=I+1
   X=(A+B)/2.
   EX=EXP(-X)
   SI=SIN(PI*X/2.)
   F=EX-SI
   WRITE(2,99)I,X,SI,EX,F
99 FORMAT(1X,I5,3F15.8,F20.8//)
   IF(F)12,10,11
12 IF(F+EPS)3,10,10
11 IF(F-EPS)10,10,3
3  IF(F*FA)5,8,6
5  B=X
   GOTO7
6  A=X
   FA=F
   GOTO7
8  WRITE(2,9)
9  FORMAT(14HA IS TRUE ROOT)
10 STOP
   END
   A>F80 =HALF
   $MAIN

     L80
```

```
Link-80  Vers. 3.41
Copyright (C) 1980 by Microsoft
Created: 28-Dec-80

*HALF,HALF/N/
Data     0103     03AD     <  682>

 FORLIB RQUEST
-$AB     0321     -$DB     02C0
-$IO     02F8     -$I1     0306
-$INIT   01B8     -$L1     037D
-$MB     0350     -$NB     029C
-$ND     0392     -$R2     01C1
-$SB     0339     -$ST     0395
-$T1     0383     -$W2     038F
-EXP     02A8     -SIN     02CC
   16 Undefined Global(s)
28756 Bytes Free

*FORLIB/S/E

Data     0103     2152     < 8271>

 FORLIB RQUEST
19900 Bytes Free
[01B4   2152        33]

A>HALF
.0          1.0          0.0001

    I          X(I)          SIN          EXP          F(X)

    1      .50000000    .70710677    .60653067    -.10057610

    2      .25000000    .38268346    .77880079     .39611733

    3      .37500000    .55557024    .68728930     .13171905
```

```
 4 .43750000       .63439322       .64564854  .01125532

 5 .46875000       .67155892       .62578404 -.04577488

 6 .45312500       .65317279       .63563871 -.01753408

 7 .44531250       .64383149       .64062405 -.00320745

 8 .44140625       .63912439       .64313138  .00400698

 9 .44335938       .64148098       .64187652  .00039554

10 .44433594       .64265698       .64125001 -.00140697

11 .44384766       .64206916       .64156318 -.00050598

12 .44360352       .64177513       .64171982 -.00005531
STOP
```

§8.2 矩陣代數與聯立方程式

§8.2.1 矩陣之基本運算

已知矩陣A與矩陣B均為3 × 3階之矩陣，即

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix},\ B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}$$

則將A與B相加，亦得一3×3階之矩陣C，C中之每一元素，爲A與B中相對應元素相加而得。

矩陣減法亦以同樣方式處理。

若將A與B相乘，則得一3×3階之矩陣P：

$$P = \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix}$$

其中：

$$\begin{aligned} p_{11} &= a_{11} \cdot b_{11} + a_{12} \cdot b_{21} + a_{13} \cdot b_{31} \\ p_{12} &= a_{11} \cdot b_{12} + a_{12} \cdot b_{22} + a_{13} \cdot b_{32} \\ p_{13} &= a_{11} \cdot b_{13} + a_{12} \cdot b_{23} + a_{13} \cdot b_{33} \\ p_{21} &= a_{21} \cdot b_{11} + a_{22} \cdot b_{21} + a_{23} \cdot b_{31} \\ &\vdots \end{aligned}$$

p中各元素P_{ij}，可寫成下列通式：

$$P_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + a_{i3}b_{3j} = \sum_{k=1}^{3} a_{ik}b_{kj} \qquad (4)$$

例：

$$\begin{pmatrix} 1 & 4 & -1 \\ 3 & 2 & 2 \end{pmatrix} \begin{bmatrix} 2 & 1 \\ 3 & 6 \\ 4 & 5 \end{bmatrix} = \begin{pmatrix} 10 & 20 \\ 20 & 25 \end{pmatrix}$$