◎ 技術報告

格點에 依해 形成되는 行列方程式의 小型 計算機에 의한 解法

李 根 伯*

—차 례—

1. 序 論

우리는 理工界의 여러 分野에서 종종 行列界에 接하게 되며, 이의 解를 求하기 위하여 聯立方程式을 풀지 않으면 안되는 경우가 許多하게 많다.

이러한 聯立方程式은 次元數가 커짐에 따라 計算量이 指數的으로 增加하기 때문에 많은 計算時間을 소모하게 된다. 電子計算機 以前의 時代, 즉 手作業이나 卓上計算機에만 依存하던 時代에는 計算量을 감소시킬 수 있는 數値計算法을 開發하기 위한 많은 硏究가 행하여졌으며, 또 可能한 限 方程式의 次數를 줄이려고 努力하였다. 以後 電子計算機가 등장하면서 이러한 問題點들이 漸次 해소되었으나, 完全히 解決되지는 않았다.

問題는 電子計算機에도 記憶容量에 限界가 있다는 點이다. 물론 補助裝置(Tape, Disk, Drum 등)를 利用하는 경우, 記憶容量을 거의 無限에 가깝도록 增加시킬 수는 있으나 이렇게 되면 計算速度가 대우 느려지기 때문에 電子計算機 使用者는 많은 經費를 所要하지 않으면 안되게 된다. 電子計算機의 보조장치는 人間에 비유한다면 노트나 메모지에 해당하는 것이므로, 전자계산기의 보조장치를 이용하는 것은 사람이 노트에 적어가면서 計算을 해나가는 것과 同一하다고 볼 수 있다. 그러므로 머리(頭腦)속에서 暗算에 의해 計算하는 경우와 比較하여 볼 때, 計算速度가 느려지게 됨은 당연한 일이라 하겠다.

本 稿의 目的은 特殊한 形態를 가지고 있는 行列界의 解를 求하는데 있어, 電子計算機가 記憶하여야만 할 部分을 可能한 한 줄임으로써 次元數가 높은 行列方程式도 電子計算機의 머리(Core)만 使用하여 풀 수 있는 몇가지 方法을 開發하여 電子計算機 使用時間과 經費를 節約할 수 있도록 하는데 있다. 여기서 말하는 特殊한 形態의 行列界란 對角線 要素에 對하여 對稱이거나 또는 要素에 0이 많이 포함되어 있는 行列界를 뜻하며, 이러한 形態가 나타나는 代表的인 例는 電氣의 回路, 上水道의 配水管網, 化學物質의 結晶構造, 構造物의 應力解析 等이며, 그 밖에 有限要素法(이 경우는 各 要素가 三角形으로 이루어져 있을 때 特히 効果的이다)을 使用하여 어떤 問題를 解決할 때도 나타나게 된다.

*서울大學校 電子計算所 助敎

2. 一般的인 行列方程式의 解法

本 章에서는 뒤에 說明될 特殊한 解法의 理解를 쉽게하고, 그 有効性을 인식시키고자 一般的인 解法, 즉 Gauss의 消去法, Gauss-Seidel의 反復法에 對해서 간략하게 說明하기로 한다. 이에 앞서 여기에 소개되는 모든 전자계산 프로그램은 서울工大 電子計算所에 설치되어 있는 IBM 1130, 16k를 보조장치(Disk)의 使用없이 利用할 수 있도록 作成되었으며, 使用言語는 FORTRAN IV임을 밝혀 둔다.

1) Gauss의 消去法

Gauss의 消去法을 간략히 說明하면, 表—①의 行列方程式을 表—②와 같이 변형시킨 다음 後方代入(Backward Substitution)에 의해 解를 求하는 方法이다.

$$\begin{pmatrix} a_{11} & a_{12}\cdots\cdots\cdots\cdots a_{1n} \\ a_{21} & a_{22}\cdots\cdots\cdots\cdots a_{2n} \\ \cdots\cdots & \cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n1} & a_{n2}\cdots\cdots\cdots\cdots a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} C_1 \\ C_2 \\ \cdots \\ C_n \end{pmatrix} \cdots 式(1)$$

위의 $n\times n$ 行列方程式에서 a_{ij}는 係數行列 A의 要素이고 X_i는 未知의 列行列 X, C_i는 常數의 列行列 C의 要素이다. 이를 表로 表示하면 表—①과 같다.

表—①

x_1	x_2	x_3	……	x_n	C_i
a_{11}	a_{12}	a_{13}	……	a_{1n}	C_1
a_{21}	a_{22}	a_{23}	……	a_{2n}	C_2
…	…	…	……	…	…
a_{n1}	a_{n2}	a_{n3}	……	a_{nn}	C_n

위의 表를 1回 1個의 未知數의 要素를 除去해 나가면서 三角形 行列로 만들면 表—②와 같이 된다.

表—②에서 먼저 $x_n=k_n$에서 x_n을 구하고, 다음 x_n을 代入하여 x_{n-1}을 구하고 계속하여 x_{n-2}, $x_{n-3}\cdots\cdots$

x_2, x_1을 求할 수 있게 된다.

表—②

x_1	x_2	x_3	……	x_n	C_i
1	t_{12}	t_{13}	……	t_{1n}	k_1
0	1	t_{23}	……	t_{2n}	k_2
0	0	0	……	…	…
0	0	0	……	1	k_n

이러한 過程을 電子計算 프로그램으로 작성하면 다음과 같으며, 이에 使用된 變數들의 內容은

E(I, J) : 表—①의 a_{ij} 및 表—②의 t_{ij}
C(I) : 表—①의 c_i 및 表—②의 k_i
B(I) : x_i
N : 行列方程式의 次元數

```
      DIMENSION E(50,
      50), C(50), B(50)
      READ(2,10)N
   10 FORMAT(15)
      READ(2,20) (E(I,J),J=1,N),I=1,N)
   20 FORMAT(20F4.0)
      READ(2,20) (C(I),I=1,N)
      M=N-1
      DO 50 I=1,M
      L=I+1
      DO 50 J=L,N
      IF(E(J,I)) 30,50,30
   30 DO 40 K=L,N
   40 E(J,K)=E(J,K)-E(I,K)*E(J,I)/E(I,I)
      C(J)=C(J)-C(I)*E(J,I)/E(I,I)
   50 CONTINUE
      B(N)=C(N)/E(N,N)
      DO 70 I=1,M
      K=N-I
      L=K+1
      DO 60 J=L,N
   60 C(K)=C(K)-B(J)*A(K,J)
   70 B(K)=C(K)/E(K,K)
      WRITE(3,80) (B(I),I=1,N)
   80 FORMAT (1HO,10(E10.4,1X))
      CALL EXIT
      END
```

上記 프로그램에서 볼 수 있듯이 50×50의 行列方程式을 풀기 위하여는 約 5.6K Words의 記憶容量이 必要하게 되며, 16K Words의 IBM 1130으로는 最大 80×80 정도의 行列方程式을 풀 수 있을 것이다. 그러나, 우리는 이 보다 훨씬 큰 거대한 行列方程式을 다루지 않으면 않될 경우를 종종 당하게 되며, 특히 어떤 문제의 解決을 위하여 有限要素法을 사용하는 경우 要素의 數를 너무 줄이게 되면 計算의 意義가 없어지게 되므로 次元數가 큰 行列界를 많이 接하게 된다. 이렇게 되면 通常 보존장치를 사용하여 計算을 行하게 되는데 앞서 말한 바와 많은 計算時間과 經費를 소모하게 된다.

2) Gauss-Seidel의 反復法

이 方法은 使用에 앞서 行列의 對角線 要素의 절대값이 그 行의 다른 要素들의 절대값의 合보다 크거나 적어도 같지 않으면 안된다. 이때 對角線 要素의 절대값이 크면 클수록 수렴 속도가 빨라지게 된다. 만일 對角線要素의 절대값이 그 行의 다른 要素들의 절대값의 合보다 작으면 수렴이 되지 않으므로 이 方法을 사용할 수 없다.

즉, 式 (1)에서

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \quad \cdots\cdots\cdots\text{式}(2)$$

의 條件이 成立되어야 한다.

Gauss-Seidel 反復法의 計算過程을 略述하면 다음과 같다.

式 (1)의 行列方程式에서 對角線要素로 그 行의 모든 要素를 나누어 式 (3)과 같이 表示한다.

$$\left.\begin{aligned} x_1 &= k_1 - b_{12}x_2 - b_{13}x_3 - \cdots\cdots - b_{1n}x_n \\ x_2 &= k_2 - b_{21}x_1 - b_{23}x_3 - \cdots\cdots - b_{2n}x_n \\ &\cdots\cdots\cdots\cdots\cdots\cdots \\ x_n &= k_n - b_{n1}x_1 - b_n x_2 - \cdots\cdots - b_{n,n-1}\, x_{n-1} \end{aligned}\right\} \cdots\text{式 (3)}$$

이때, $k_i = c_i/a_{ii}$ $\quad i=1,2,\cdots n$

$b_{ij}=a_{ij}/a_{ii}$ $\quad j=1,2,\cdots n\,;\ i=1,2,\cdots n\,;\ i\neq j$

式 (3)에 초기가정치 $x_i^{(0)}$를 代入하여 改良値 $x_i^{(1)}$을 求하고, 다시 $x_i^{(1)}$을 代入하여 $x_i^{(2)}$를 구하고 하는 過程을 反復하여 $|x_i^{(l)}-x_i^{(l-1)}|$이 원하는 범위내에 들때까지 反復한다. 이때 $x_i^{(0)}=0$로 하여 反復計算을 시작하는 方法을 특히 Jacobi 反復法이라 부르는데 다음의 電子計算 프로그램은 이 方法을 使用한 것이다. 프로그램內의 變數의 內容은 앞의 消去法에서와 같으며 計算의 정확도를 정하는 문제는 대단히 전문적인 것으로 문제의 성격에 따라 달라져야 할 것이다.

```
      DIMENSION
      E(50,50),C(50),B(50)
      READ(2,10) N
   10 FORMAT(15)
      READ(2,20) ((E(I,J),J=1,N),I=1,N)
   20 FORMAT(20F4.0)
      READ(2,20) (C(I),I=1,N)
      DO 30 I=1,N
   30 B(I)=0.
```

```
      DO 90 L=1, 100
      TO=0.
      DO 70 I=1, N
      Y=C(I)
      DO 50 J=1, N
      IF(I-J) 40, 50, 40
40    Y=Y-E(I, J)*B(J)
50    CONTINUE
      A=Y/E(I, I)
      AB=ABS(A-B(I))
      B(I)=A
      IF(AB-TO) 70, 70, 60
60    TO=AB
70    CONTINUE
      IF(TO-0.00001) 100, 100, 90
90    CONTINUE
100   WRITE(3, 110) L, TO
110   FORMAT(1HO, 'IT ERATION TIME=',
      13, 10X, 'MAXIMUM DIFFERENCE=',
      E10.5 1)
      WRITE(3, 120) (B(I), I=1, N)
120   FORMAT(1HO, 10(E10.4, 1×))
      CALL EXIT
      END
```

3. 特殊한 形態의 行列方程式의 解法

1) 띠行列에 대한 消去法

이것은 式 (1)의 係數行列 [A]가 0이 아닌 값을 갖는 要素들이 對角線要素를 中心으로 하여 아래와 같이 띠(Band)를 이루고 있는 경우, 0의 값을 갖는 要素들을 제외한 띠 部分만 가지고 새로운 行列을 만들어 消去法을 적용하는 方法이다.

$$[A]=\begin{bmatrix} a_{11} & \cdots & & 0 \\ \vdots & a_{22} & & \vdots \\ & & a_{33} & \\ 0 & \cdots & & a_{nn} \end{bmatrix} \cdots\cdots\cdots\cdots\cdots\cdots 式\ (4)$$

위의 行列에서 띠의 幅 M은 다음에 의해 決定된다.

$$M=2L+1 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots 式\ (5)$$

여기서 L은 한行 中에서 0이 아닌 값을 갖는 요소간의 最大間隔이다.

式 (4)의 行列$[A]$의 변형을 行列$[A_s]$라 하면

$$[A_s]=\left(\begin{array}{c|c} 0\cdots\cdots 0 & a_{1,\ (M+1)/2}\cdots\cdots a_{1n} \\ \cdots\cdots\cdots\cdots & \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{nM}\ \cdots\cdots & a_{n1(M+1)/2}\ \ 0\cdots 0 \end{array}\right) \cdots\cdots 式\ (6)$$

과 같이 된다.

行列$[A_s]$에 대한 消去法의 電子計算 프로그램과 이에 使用된 變數의 內容은 다음과 같다.

E(I, J) : $[A_s]$의 要素

C(I) : 式 (1)의 常數行列 및 未知數行列

N : 行列 $[A]$의 次元數

M : 띠의 幅

MC : 對角線要素를 포함한 띠의 반쪽 幅

이 경우 특히 行列 $[A]$가 對稱인 경우는 式 (6)의 行列 $[A_s]$의 쇄선(⁞) 오른쪽 部分만 E(I, J)에 한 行에 MC個만큼 記憶시키면 되므로 記憶容量을 상당히 절약할 수 있게 된다.

```
      DIMENSION E(100, 150), C(100)
      READ(2, 10) N, M
10    FORMAT(215)
      READ(2, 20) ((E(I, J), J=1, M), I=1, N)
20    FORMAT(20F4.0)
      READ(2, 20) (C(I), I=1, N)
      MC=(M+1)/2
      N1=N-1
      DO 70 I=1, N1
      C(I)=C(I)/E(I, MC)
      DO 30 JJ=2, MC
      J=JJ+MC-1
30    E(I, J)=E(I, J)/E(I, MC)
      DO 60 K=2, MC
      JJ=MC-K+1
      II=I+K-1
      IF(N-II) 70, 40, 40
40    J=MC-K+1
      DO 50 L=2, MC
      J=J+1
      LL=L+MC-1
50    E(II, J)=E(II, J)-E(II, JJ)*E(I, LL)
60    C(II)=C(II)-C(I)*E(II, JJ)
70    CONTINUE
      C(N)=C(N)/E(N,MC)
      DO 90 J=1, N1
      DO 80 K=2, MC
      KK=MC+K-1
      LL=N-J+K-1
      I=N-J
      C(I)=C(I)-E(I, KK)*C(LL)
80    CONTINUE
90    CONTINUE
```

```
 95 FORMAT(1HO, 10E11.4)
    WRITE(3,95) (C(I),I=1,N)
    CALL EXIT
    END
```

2) 反復法의 변형

이 方法은 行列要素들의 大部分이 0인 경우 매우 効果的인 것으로 各 行의 0이 아닌 要素의 값과 그 列番號를 電子計算機에 記憶시켜 反復法에 의해 計算하는 方法이다.

이의 電子計算 프로그램과 프로그램에 사용되는 變數의 內容은 다음과 같다.

MC(I,J) : I 行에서 0이 아닌 값을 갖는 J번째 列의 番號(다음 節의 表—④ 참조)

E(I,J) : I 行에서 0이 아닌 값을 갖는 J번째 列의 값

C(I) : 式 (1)의 C_i

D(I) : 式 (1)의 係數行列의 對角線要素 즉 a_{ii}

B(I) : 式 (1)의 未知數行列

N : 次元數

M : 한 行에서 0이 아닌 값을 갖는 最大數

L : 反復回數

TO : $|x_i^{(l)}-x_i^{(l-1)}|$중의 最小値

여기서 주의할 것은 對角線要素는 D(I)에 記憶시키면 되고, MC(I,J), E(I,J)에는 記憶시킬 필요가 없다는 점이다.

```
      DIMENSION MC(100,6), E(100,6),
      D(100), C(100), B(100)
      READ(2,10) M,N
  10  FORMAT(2I5)
      READ(2,20) ((MC(I,J), J=1,M), I=1,N)
  20  FORMAT(20I4)
      READ(2,30) ((E(I,J), J=1,M), I=1,N)
  30  FORMAT(20F4.0)
      READ(2,30) (D(I), I=1,N)
      READ(2,30) (C(I),I=1,N)
      DO 40 I=1,N
  40  B(I)=0.
      DO 100 L=1, 100
      TO=0.
      DO 80 I=1,N
      Y=C(I)
      DO 60 J=1,M
      K=MC(I,J)
      IF(K) 70,70,50
  50  Y=Y+E(I,J)*B(K)
  60  CONTINUE
  70  A=Y/D(I)
      AB=ABS(A-B(I))
      B(I)=A
      IF(AB-TO) 80,80,75
  75  TO=AB
  80  CONTINUE
      IF(TO-0.00001) 200,200,100
 100  CONTINUE
 200  WRITE(3,85) L, TO
  85  FORMAT(1HO, 'ITERATION TIME=',
      I3,10×, 'MAXIMUM DIFFERENCE=',
      E10.5 1)
      WRITE(3,90) (B(II),II=1,N)
  90  FORMAT(1HO,10(E10.4,1×))
      CALL EXIT
      END
```

3) 還元法

이 方法은 聯立方程式을 준다는 근본적인 點에서는 演算回數가 너무 많아지기 때문에 實用的이 아니다. 그러나, 係數가 0인 要素가 많은 行列方程式의 경우 이 還元法에 의해 行列의 次元數를 감소시킨 다음, 次元數가 감소된 즉 還元된 行列을 消去法에 의해 풀면, 작은 記憶容量으로 큰 次元數의 行列方程式을 다루는 것이 可能하여진다.

설명을 간단히 하기 위하여 式 (7)의 2元聯立 1次方程式을 例로 還元法의 原理를 기술하기로 한다.

$$\left.\begin{array}{l}ax+by=c\\dx+ey=f\end{array}\right\}\cdots\cdots\text{式 (7)}$$

임의의 값 $x=x_0$에 대하여 式 (7)의 第1式에서 y를 구하면

$$y=(c-ax_0)/b \quad (b\fallingdotseq 0)$$

로 된다($b=0$의 경우는 $a\fallingdotseq 0$으로 되므로 먼저 $y=y_0$를 택하여 x를 구하면 된다). 얻어진 y의 값을 第2式에 代入하여 (左邊—右邊)의 값을 $G(x_0)$라 하면

$$G(x_0)=\{(bd-ae)x_0+ce-bf\}/b$$

로 된다. 여기에서 x의 참값을 $x=x$라 하면 다음의 관계가 成立한다.

$$G(x)\times G(x_0)=\left(\frac{dG}{dx}\right)_{x=x_0}(x-x_0)\cdots\cdots\text{式 (8)}$$

式 (8)은 一般的인 경우에는 近似式이 되지만, 聯立 1次方程式에 對해서는 $G(x)$가 x의 1차함수이므로 이 關係에는 誤差가 포함되지 않는다. 마찬가지로 다음의 關係에도 誤差는 포함되지 않는다.

$$\left(\frac{dG}{dx}\right)_{x=x_0}=\frac{G(x_1)-G(x_0)}{x_1-x_0}=\frac{bd-ae}{b}\text{式 (9)}$$

참값 x에 對하여 $G(x)=0$으로 되지 않으면 안되므

로 x로 表示된 式 (8)에 式 (9)를 代入하면

$$x=\left\{\left(\frac{dG}{dx}\right)_{x=x_0}x_0-G(x_0)\right\}\Big/\left(\frac{dG}{dx}\right)_{x=x_0}=\frac{bf-ce}{bd-ae}$$

가 얻어지며, 이를 式 (7)에 代入하여 y를 求하면 x, y가 모두 얻어지게 된다.

式 (10)의 3元聯立 1次方程式에 對한 還元法은 아래와 같다.

$$\left.\begin{aligned} ax+by+cz&=d\\ ex+fy+gz&=h\\ ix+jy+kz&=l \end{aligned}\right\}\cdots\cdots\cdots\cdots\text{式 (10)}$$

$c\neq0$이라도 一般性을 상실한 것은 아니므로 $c\neq0$로 한다. $(x=x_0,\ y=y_0)$란 임의의 값에 對하여 第1式에서 Z를구한다. 얻어진 값을 第2式과 第3式에 代入하여 (左邊一右邊)의 값을 計算하여, 各各을 $G(x_0, y_0)$, $H(x_0, y_0)$라 한다. 다음, $(x=x_1,\ y=y_0)$, $(x=x_0,\ y=y_1)$에 의한 값(x_1, y_1은 임의의 값이나, $x_1\neq x_0$, $y_1\neq y_0$로 한다)에 對하여, 각각 $G(x_1, y_0)$, $H(x_1, y_0)$와 $G(x_0, y_1)$, $H(x_0, y_1)$을 구하여 다음 式과 같이 全微分形式

$$\left.\begin{aligned} \frac{\partial G}{\partial x}dx+\frac{\partial G}{\partial y}dy&=dG\\ \frac{\partial H}{\partial x}dx+\frac{\partial H}{\partial y}dy&=dH \end{aligned}\right\}\cdots\cdots\cdots\text{式 (11)}$$

과 같이 表示하면

$$\left.\begin{aligned} &\frac{G(x_1,\ y_0)-G(x_0,\ y_0)}{x_1-x_0}(x-x_0)\\ &\quad+\frac{G(x_0,\ y_1)-G(x_0,\ y_0)}{y_1-y_0}(y-y_0)\\ &=-G(x_0,\ y_0)\\ &\frac{H(x_1,\ y_0)-H(x_0,\ y_0)}{x_1-x_0}(x-x_0)\\ &\quad+\frac{H(x_0,\ y_1)-H(x_0,\ y_0)}{y_1-y_0}(y-y_0)\\ &=-H(x_0,\ y_0) \end{aligned}\right\}\cdots\text{式 (12)}$$

로 되어, 還元된 2元聯立 1次方程式이 된다. 式 (12)에 對하여 다시 式 (7)에 對하여 行하였던 것과 同一한 조작을 하여 풀면 x와 y가 얻어지고, 그 값을 式 (10)의 어느 式에 代入하여 z를 구하면 결국 式 (10)을 만족시키는 解 x, y, z를 얻게 된다.

일반적으로 어떠한 行列方程式도 還元시킬 수는 있으나, 演算回數가 너무나 많아지기 때문에 實用的이 아니다. 그러나, 係數가 0인 要素를 많이 가지고 있는 行列方程式에서는 매우 効果的이다.

表—③ 6×6 行列方程式의 係數表

行番號	1	2	3	4	5	6	常數項
1	3.0	1.0	−1.0	0.0	0.0	0.0	2.0
2	2.0	2.0	0.0	0.0	1.0	0.0	1.0
3	0.0	1.0	2.0	−1.0	0.0	0.0	−3.0
4	0.0	0.0	3.0	−2.0	4.0	4.0	−8.0
5	1.0	0.0	0.0	3.0	0.0	1.0	5.0
6	1.0	1.0	0.0	0.0	1.0	1.0	−1.0

그러면, 表—③과 같이 係數가 0인 要素를 많이 포함하고 있는 6×6 行列方程式에 對하여 還元法을 적용하여 보자.

還元過程을 위하여 各 行에서 係數가 0이 아닌 列의 番號를 적은 還元過程表(表—④)를 작성한다.

表—④ 還元過程表

行番號	係數가 0이 아닌 列의 番號				順序
1	1	2	3		
2	1	2	5		
3	2	3	4		
4	3	4	5	6	
5	1	4	6		
6	1	2	5	6	

表—④에 對한 還元過程을 順序대로 記述하면 다음과 같다.

(1) 1行에서 1을 ○표로 둘러싸고, 2~6行까지에 나오는 모든 1에 ○표를 한다.

(2) 1行에서 2를 ○표로 둘러싸고, 2~6行까지의 모든 2에 ○표를 한다.

(3) 1行의 3을 □표로 둘러싸고, 順序란에 숫자 1을 記入한다. 이것은 第1式에서 x_1과 x_2를 구하면 x_3를 구할 수 있다는 것을 뜻한다. 다음, 2~6行까지의 모든 3에 ○표를 한다.

(4) 2行에 있어 5를 제외한 1과 2는 이미 ○표로 둘러싸여 있다. 즉 x_1과 x_2의 값이 정하여지면 x_5의 값은 곧 決定된다. 따라서 5에는 □표를 한다. 그리고, 順序란에 숫자 2를 記入하고 3~6行에 나오는 모든 5에 ○표를 한다.

(5) 똑같은 操作을 3行의 4에 對하여 行한다.

(6) 4行의 6을 □표로 둘러싸고 순서란에 숫자 4를 記入한다. 5, 6行에서 6에 ○표를 하여야 하나 5, 6行에 있어서는 x_6 以外에는 未知數가 없으므로 6마저 ○표를 하고 나면 어떤 표로도 둘러싸여 있지 않은 列番號는 하나도 없게 된다. 이런 경우에는 순서란에 로마 숫

表—⑤ 還元結果表

行番號	係數가 0이 아닌 列의 番號				順序
1	①	②	$\boxed{3}$		1
2	①	②	$\boxed{5}$		2
3	②	③	$\boxed{4}$		3
4	③	④	⑤	$\boxed{6}$	4
5	①	④	6		I
6	①	②	⑤	6	II

자를 Ⅰ, Ⅱ, …의 순으로 기입한다.

以上의 結果가 表－⑤에 나타나 있다.

위의 表－⑤ 中에서 임의의 값을 주어야 하는 未知數는 ◎表로 둘러싸여 있는데, 이 경우는 x_1, x_2에 해당된다. 또 各 式에서 그 값을 計算하여야 할 未知數에는 □表가 되어 있는데, 예를 들면 第1式에서 x_3, 第2式에서 x_5 等 순서란에 번호 순서대로 計算하면 된다. 5, 6行의 순서란에는 로마숫자가 기입되어 있는데, 第5式과 第6式에서 (左邊－右邊)의 값을 구하게 된다.

◎表 數와 순서란의 로마숫자의 最大値는 반드시 일치하여야 하며, 그 수는 還元된 方程式의 次元數를 뜻한다. 그러므로, 이번 例에서는 6元이 2元으로 還元된 것이다.

還元된 行列方程式을 作成하는 과정은 表－⑤에 따라 기계적으로 행하면 된다.

$x_1=0$, $x_2=0$으로 하면 第1式에서 $x_3=-2$, 第2式에서 $x_5=1$, 第3式에서 $x_4=-1$, 第4式에서 $x_6=-2$가 얻어진다. 第5式과 第6式에 對하여 (左邊－右邊)의 값을 각각 G, H라 하면, 第5式에서 $G(x_1=0, x_2=0)=-10$, 第6式에서 $H(x_1=0,\ x_2=0)=0$으로 된다.

$x_1=1$, $x_2=0$으로 하면, 마찬가지로 $G(1,0)=25/4$, $H(1,0)=7/4$가 얻어지며, x_1 0, $x_2=1$에 對하여는 $G(0,1)=-15/4$, $H(0,1)=7/4$로 된다.

計算過程을 表로 作成하면 表－⑥과 같으며, 이 경우 還元聯立 1次方程式은 그 次元이 2로 되어 式 (12)와 같은 모양으로 되어 最終으로 다음의 式 (13)으로 된다.

$$\left.\begin{array}{r} 13x_1+5x_2=8 \\ x_1+x_2=0 \end{array}\right\} \quad \cdots\cdots\cdots\cdots\text{式 (13)}$$

x_3, x_4, x_5, x_6를 구하기 위하여는 式 (13)을 풀어서 $x_1=1$, $x_2=-1$을 구한 다음, 다시 한번 上述한 過程을 반복하면 된다. 그 結果가 表－⑥의 오른쪽 끝에 나와 있다.

表－⑥ 計 算 過 程

順序	式番號	x_1 x_2	0 0	1 0	0 1	1 -1
1	1	x_3	-2	1	-1	0
2	2	x_4	1	-1	-1	1
3	3	x_5	-1	5	2	2
4	4	x_6	-2	3/4	3/4	-2
Ⅰ	5	G	-10	25/4	$-15/4$	
				(1, 0)－(0, 1)	(1, 0)－(0, 0)	
				65/4	25/4	
Ⅱ	6	H	0	7/4	7/4	
				(1. 0)－(0, 0)	(0. 1)－(0, 0)	
				7/4	7/4	

以上의 모든 過程에 대한 電子計算 프로그램을 모두 本稿에 실렸으면 하나, 부피관계상 表－⑤에서 表－⑥을 만드는 還元過程의 部分만을 게재한다. 이에 使用된 중요한 變數의 內容은 다음과 같다.

- KS(I, J) : 表－④의 I行에서의 J번째 列番號
- NS(I) : 表－⑤에서 1, 2와 같이 ◎表로 둘러 싸이는 번호
- N2 : 還元方程式의 次元數
- S(I, J) : 表－⑤에서 ◎로 싸여 있으면 +1, □면 －1, ○면 0이다.

```
      DO 998 I=I, MI
      READ(2, 3) (KS(I, J), J=1, N3)
998   CONTINUE
      DO 410 I=1, MI
      Q(I)=1.
      IX=0
      DO 408 J=1, N3
      K=KS(I, J)
      IF(K) 409, 409, 402
402   S(I, J)=1.
      IX=IX+1
408   CONTINUE
409   F(I)=FLOAT(IX)+0.1
410   CONTINUE
      N2=0
      IX=0
      IY=0
      KEY=0
      KK=1
419   M=KNOWN(KK)
411   DO 420 I=1, MI
      IF(Q(I)) 420, 495, 412
412   DO 418 J=1, N3
      K=KS(I, J)
      IF(K) 420, 420, 415
415   IF(M-K) 418, 416, 418
416   S(I, J)=-1.
      F(I)=F(I)-1.
      GO TO 420
418   CONTINUE
420   CONTINUE
      IF (KEY) 417, 417, 750
417   KK=KK+1
      IF (KK-KN) 419, 419, 750
750   BX=100.
```

```
      DO 425 I=1, MI
      IF(Q(I)) 425, 495, 421
421   IF(F(I)-BX) 422, 425, 425
422   BX=F(I)
425   CONTINUE
      BX=BX+0.4
      DO 430 I=1, MI
      IF(Q(I)) 430, 495, 428
428   IF(F(I)-BX) 432, 430, 430
430   CONTINUE
432   M=I
      IF(BX-1.) 440, 459, 459
440   DO 450 I=M, MI
      IF(Q(I)) 450, 495, 442
442   IF(F(I)-1.) 443, 450, 450
443   L=MI-IY
      D(L)=I
      IY=IY+1
      Q(I)=-1.
450   CONTINUE
      IF (IY+IX-MI) 750, 455, 495
455   DO 458 I=1, MI
      IF(Q(I)) 458, 495, 459
458   CONTINUE
      GO TO 490
459   IF(BX-2.) 460, 480, 480
460   DO465 J=1, N3
      IF(S(M, J)) 465, 495, 461
461   Q(M=-1.
      IX=IX+1
      D(IX)=M
      L=KS(M, J)
      KS(M, J)=-L
      M=L
      KEY=KEY+1
      GO TO 411
465   CONTINUE
480   DO 485 J=1, N3
      IF(S(M, J) 485, 495, 481
481   N2=N2+1
      M=KS(M, J)
      NS(N2)=M
      KEY=KEY+1
      GO TO 411
485   CONTINUE
490   DO 491 I=1, MI
      F(I)=KS(I, N3)
      DO 491 J=1, N1
491   S(I, J)=KS(I, J)
      DO 493 I=1, MI
      K=D(I)
      DO 492 J=1, N1
492   KS(I, J)=S(K, J)
493   KS(I, N3)=F(K)
      GO TO 497
495   WRITE (3, 496)
496   FORMAT (/5X, 'ERROR')
      GO TO 800
497   WRITE (3, 498) N2, (NS(I), I=1, N2)
498   FORMAT(//I5, 10X, 20I5)
      DO 1000 I=1, MI
1000  WRITE (3, 3) (KS(I, J), J=1, N3)
      IF(N2-M5) 499, 499, 702
702   WRITE (3, 703)
703   FORMAT(/5X, 'N2 IS LARGE')
      GO TO 800
```

4. 結　　言

以上의 3個 方法에 對하여 電氣回路와 흡사한 성격을 지닌 上水道의 配水管網, 有限要素法에 의한 熱傳達解析, 流體의 壓力 및 速度分布 等에 應用해본 결과 위에 列擧한 問題들과 같이 格點(Node)에 의해 行列이 만들어지는 경우 매우 効果的인 것으로 밝혀졌다. 특히, 行列이 對稱行列인 경우에는 各 格點의 番號를 주의깊게 정하여 띠(Band)의 幅을 줄이면 記憶容量도 상당히 節約될 뿐 아니라, 3個 方法 중 計算速度도 가장 빨랐다. 그러나 格點番號를 잘못 정하면 反對가 記憶容量을 많이 소모하게 되어 補助 장치를 사용하지 않으면 안되는 不便이 있었다.

두번째의 반복법을 변형시킨 方法은 모든 行의 對角線要素가 그 行의 다른 要素의 合보다 매우 크게 되면(式 (2)), 수렴이 빨라지기 때문에 計算速度도 比較的 빨라지고 格點의 番號를 定하는데 전혀 신경쓸 필요가 없으므로, 回路, 配水管網, 構造物의 解析 및 要素가 三角形인 有限要素法에 응용하면 매우 効果的일 것이다. 그러나 要素가 三角形이 아니면 式 (2)를 만족시키지 못하므로 使用이 不可能하다.

세번째의 還元法은 전혀 制約條件은 없으나 係數의 값이 0인 要素가 적으면 計算時間이 너무 오래 걸리므로 實用的이 아니다. 格點數가 220點이고 各 格點에 接續되는 格點數가 最大 5個인, 다시말하면 220×220의

〈p. 21 계속〉