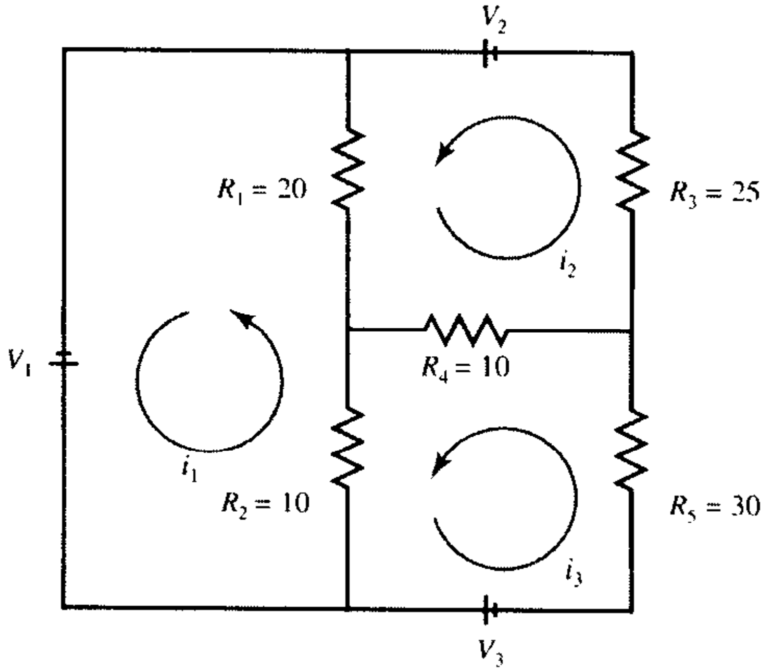
Lecture 4

第三章 解線性方程組：直接法

方程組

- 方程組
  - *線性方程組(Systems of Linear Equations)*
  - *非線性方程組(Systems of Non-linear Equations)*
- 方程組解法
  - *直接法(Direct Methods)(第3章)*
  - *迭代法(Iterative Methods)(第6章)*

問題一：電路分析



問題一：電路分析

左側迴路：$20(i_1 - i_2) + 10(i_1 - i_3) = 0$

右上迴路：$25i_2 + 10(i_2 - i_3) + 20(i_2 - i_1) = 0$

右下迴路：$30i_3 + 10(i_3 - i_2) + 10(i_3 - i_1) = 200$

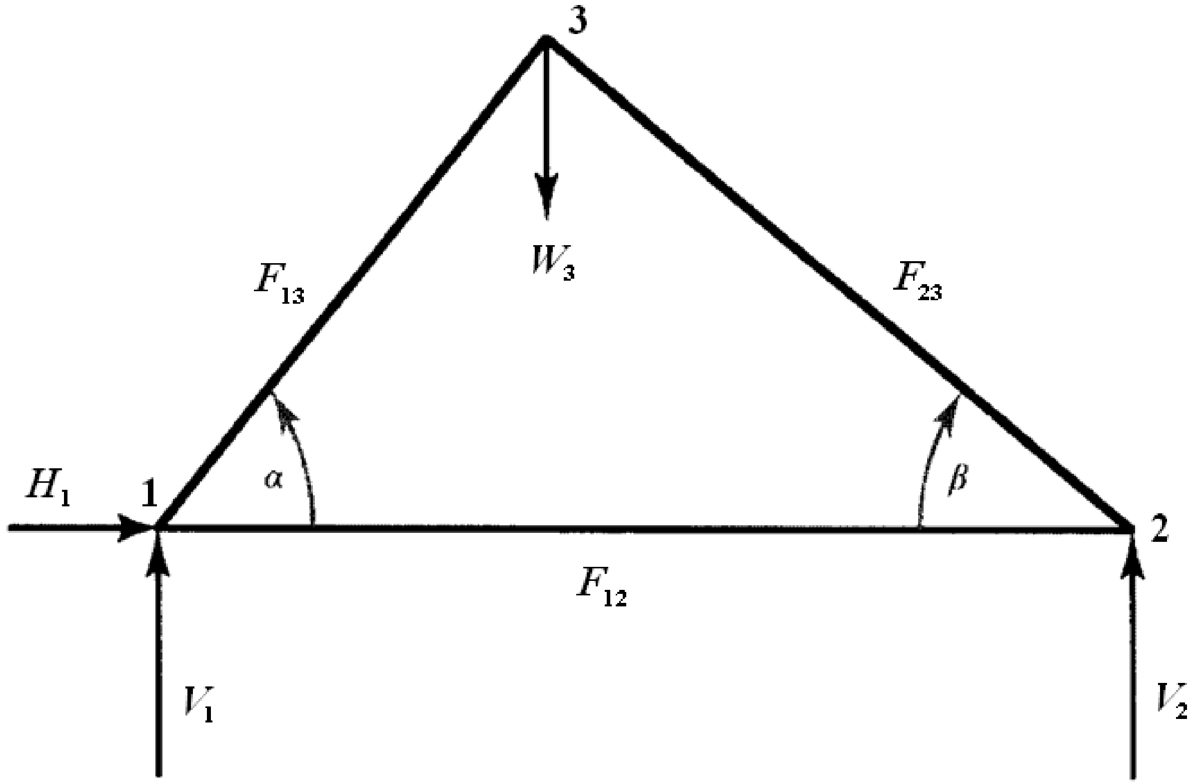
化簡成線性方程組

$$30i_1 - 20i_2 - 10i_3 = 0$$

$$-20i_1 + 55i_2 - 10i_3 = 0$$

$$-10i_1 - 10i_2 + 50i_3 = 200$$

問題二：桁架的力學分析



問題二：桁架的力學分析

節點1

$$V_1 + F_{13}\sin(\alpha) = 0$$

$$H_1 + F_{12} + F_{13}\cos(\alpha) = 0$$

節點2

$$V_2 + F_{23}\sin(\beta) = 0$$

$$-F_{12} - F_{23}\cos(\beta) = 0$$

節點3

$$-F_{13}\sin(\alpha) - F_{23}\sin(\beta) = W_3$$

$$-F_{13}\cos(\alpha) + F_{23}\cos(\beta) = 0$$

大綱

- 3-1 高斯消去法(Gaussian Elimination)
- 3-2 高斯-喬丹法(Gauss-Jordan)
- 3-3 三對角線方程組(Tridiagonal Systems)
- 3-4 MATLAB內建函數

高斯消去法(Gaussian Elimination)

- 基本觀念是將原方程組，轉換成一個同義，但較容易求解的方程組。

高斯消去法(Gaussian Elimination)

- 基本方法
- 列樞軸變換(Row Pivoting)

高斯消去法(Gaussian Elimination)

- 在高斯消去法中，原方程組被轉換成上三角方程組，然後以後向代換求解。
- 基本高斯消去法的程序：
  - *將一列乘以某數後加到第二列，以改變第二列在指定的位置之係數出現零。*

例題3.1 3x3方程組

- 以基本高斯消去法求解：

$$x+2y+3z=1 \qquad (3.1)$$

$$2x+6y+10z=0 \qquad (3.2)$$

$$3x+14y+28z=-8 \qquad (3.3)$$

例題3.1 3x3方程組

- 步驟1：利用第一式，消去第二及第三式中的x。
  - *將(3.1)式乘-2再加到(3.2)式以得到新的第二式。*
  - *將(3.1)式乘-3再加到(3.3)式以得到新的第三式*

$$x+2y+3z=1 \qquad (3.4)$$

$$2y+4z=-2 \qquad (3.5)$$

$$8y+19z=-11 \qquad (3.6)$$

例題3.1 3x3方程組

- 步驟2：利用第二式以消去第三式中的y。
  - *將 (3.5)式乘-4再加到(3.6)式，以得到新的第三式。*

$$x + 2y + 3z = 1$$

$$2y + 4z = -2$$

$$3z = -3$$

例題3.1 3x3方程組

- 步驟3:用「後向代換(Back Substitution)」來求解這個上三角方程組。

$$3z = -3 \Rightarrow z = -1$$

$$2y + 4z = -2 \Rightarrow y = 1$$

$$x + 2y + 3z = 1 \Rightarrow x = 2$$

例題3.1 3x3方程組

- 矩陣解法
- 此方程組可寫成Ax=b, 其中

$$A=\begin{bmatrix}1 & 2 & 3\\ 2 & 6 & 10\\ 3 & 14 & 28\end{bmatrix},\ \ b=\begin{bmatrix}1\\ 0\\ -8\end{bmatrix},\ \ \mathrm{x}=\begin{bmatrix}x\\ y\\ z\end{bmatrix}$$

- 將兩個矩陣合併成擴張矩陣

$$[A\ \ b]=\begin{bmatrix}1 & 2 & 3 & 1\\ 2 & 6 & 10 & 0\\ 3 & 14 & 28 & -8\end{bmatrix}$$

例題3.1 3x3方程組

- 將列$\mathbf{R_i}$乘以m之後加到列$\mathbf{R_j}$有以得到新的列$\mathbf{R_j}$，即

$$mR_i + R_j \Rightarrow R_j$$

- 第一步，將第一列乘以-2加到第二與第三列，使得擴張矩陣成為

$$\begin{bmatrix}1 & 2 & 3 & 1\\ 0 & 2 & 4 & -2\\ 0 & 8 & 19 & -11\end{bmatrix}$$

例題3.1 3x3方程組

- 第二步，將第二列乘以-4後再加到第三列，使得擴張矩陣成為

$$\begin{bmatrix} 1 & 2 & 3 & 1 \\ 0 & 2 & 4 & -2 \\ 0 & 0 & 3 & -3 \end{bmatrix}$$

- 第三步：用「後向代換(Back Substitution)」來求解這個上三角方程組。

$$3z = -3 \Rightarrow z = -1$$

$$2y + 4z = -2 \Rightarrow y = 1$$

$$x + 2y + 3z = 1 \Rightarrow x = 2$$

基本高斯消去法通式

- 有一4x4方程組可寫成Ax=b，其中

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}, \quad \mathrm{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

基本高斯消去法通式

- **步驟1：樞軸元素為a_{11}**
  - ➢ *將第一列乘上$m_{21}=-a_{21}/a_{11}$，將結果加到第二列以得到*

$$a_{21} \Rightarrow 0, a_{22} \Rightarrow a_{22} + m_{21}a_{12}, a_{23} \Rightarrow a_{23} + m_{21}a_{13}, a_{24} \Rightarrow a_{24} + m_{21}a_{14}$$
$$b_2 \Rightarrow b_2 + m_{21}b_1$$

  - ➢ *將第一列乘上$m_{31}=-a_{31}/a_{11}$，將結果加到第三列以得到*

$$a_{31} \Rightarrow 0, a_{32} \Rightarrow a_{32} + m_{31}a_{12}, a_{33} \Rightarrow a_{33} + m_{31}a_{13}, a_{34} \Rightarrow a_{34} + m_{31}a_{14}$$
$$b_3 \Rightarrow b_3 + m_{31}b_1$$

基本高斯消去法通式

  - ➢ *將第一列乘上$m_{41}=-a_{41}/a_{11}$，將結果加到第四列以得到*

$$a_{41} \Rightarrow 0, a_{42} \Rightarrow a_{42} + m_{41}a_{12}, a_{43} \Rightarrow a_{43} + m_{41}a_{13}, a_{44} \Rightarrow a_{44} + m_{41}a_{14}$$
$$b_4 \Rightarrow b_4 + m_{41}b_1$$

- **步驟2：樞軸元素為a_{22}**
  - ➢ *將第二列乘上$m_{32}=-a_{32}/a_{22}$，將結果加到第三列以得到*

$$a_{31} \Rightarrow 0, a_{32} \Rightarrow 0, a_{33} \Rightarrow a_{33} + m_{32}a_{23}, a_{34} \Rightarrow a_{34} + m_{32}a_{24}$$
$$b_3 \Rightarrow b_3 + m_{32}b_2$$

基本高斯消去法通式

➢ *將第二列乘上$m_{42}=-a_{42}/a_{22}$，將結果加到第四列以得到*

$$a_{41} \Rightarrow 0, a_{42} \Rightarrow 0, a_{43} \Rightarrow a_{43} + m_{42}a_{23}, a_{44} \Rightarrow a_{44} + m_{42}a_{24}$$

$$b_4 \Rightarrow b_4 + m_{42}b_2$$

- **步驟3：樞軸元素為a_{33}**

➢ *將第三列乘上$m_{43}=-a_{43}/a_{33}$，將結果加到第四列以得到*

$$a_{41} \Rightarrow 0, a_{42} \Rightarrow 0, a_{43} \Rightarrow 0, a_{44} \Rightarrow a_{44} + m_{43}a_{34}$$

$$b_4 \Rightarrow b_4 + m_{43}b_3$$

基本高斯消去法通式

- **步驟4：後向代換求解**

$$x_4 = \frac{b_4}{a_{44}}, x_3 = \frac{b_3 - a_{34}x_4}{a_{33}}, x_2 = \frac{b_2 - a_{23}x_3 - a_{24}x_4}{a_{22}}, x_1 = \frac{b_1 - a_{12}x_2 - a_{13}x_3 - a_{14}x_4}{a_{11}}$$

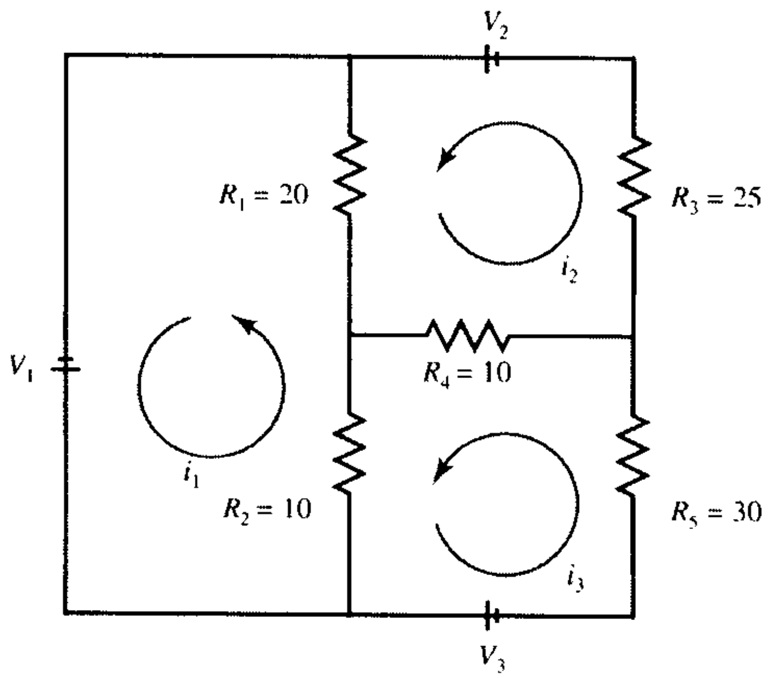
Matlab 程式

```
function x = Gauss(A, b)

% Gaussian Elimination for Linear Systems
fprintf('\n\n-------------------------------------------------\n');
fprintf('-------------- Gaussian Elimination -----------\n');

[n,m] = size(b);
fprintf('Expansion Matrix \n');
C = [ A   b ]
for i = 1 : n-1
    mm(i+1:n, i) =-C(i+1:n, i)/C(i,i);
    C(i+1:n, :) = C(i+1:n, :) +mm(i+1:n, i)*C(i, :);
end
x(n,1:m) = C(n, n+1:n+m)/C(n,n);
for i = n-1 : -1 : 1
    x(i, 1:m) = (C(i, n+1:n+m) - C(i,i+1:n)*x(i+1:n, 1:m))/C(i,i);
end
```


問題一：電路分析



問題一：電路分析

$$30i_1 - 20i_2 - 10i_3 = 0$$
$$-20i_1 + 55i_2 - 10i_3 = 0$$
$$-10i_1 - 10i_2 + 50i_3 = 200$$

$$A = \begin{bmatrix} 30 & -20 & -10 \\ -20 & 55 & -10 \\ -10 & -10 & 50 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \\ 200 \end{bmatrix}, \quad \mathrm{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

問題一：電路分析

- 主程式：

```
clear all;
close all;

A=[30 -20 -10;-20 55 -10; -10 -10 50];
b=[0; 0; 200];

tic;
x=Gauss(A,b);
Gauss_Time=toc;
fprintf('Solution: \n');
fprintf('x=[ %3.0f; %3.0f; %3.0f]',x);
fprintf('\nRun Time for Baitstow Method =%6.4f (sec)\n\n',Gauss_Time);
```

問題一：電路分析

- 結果：

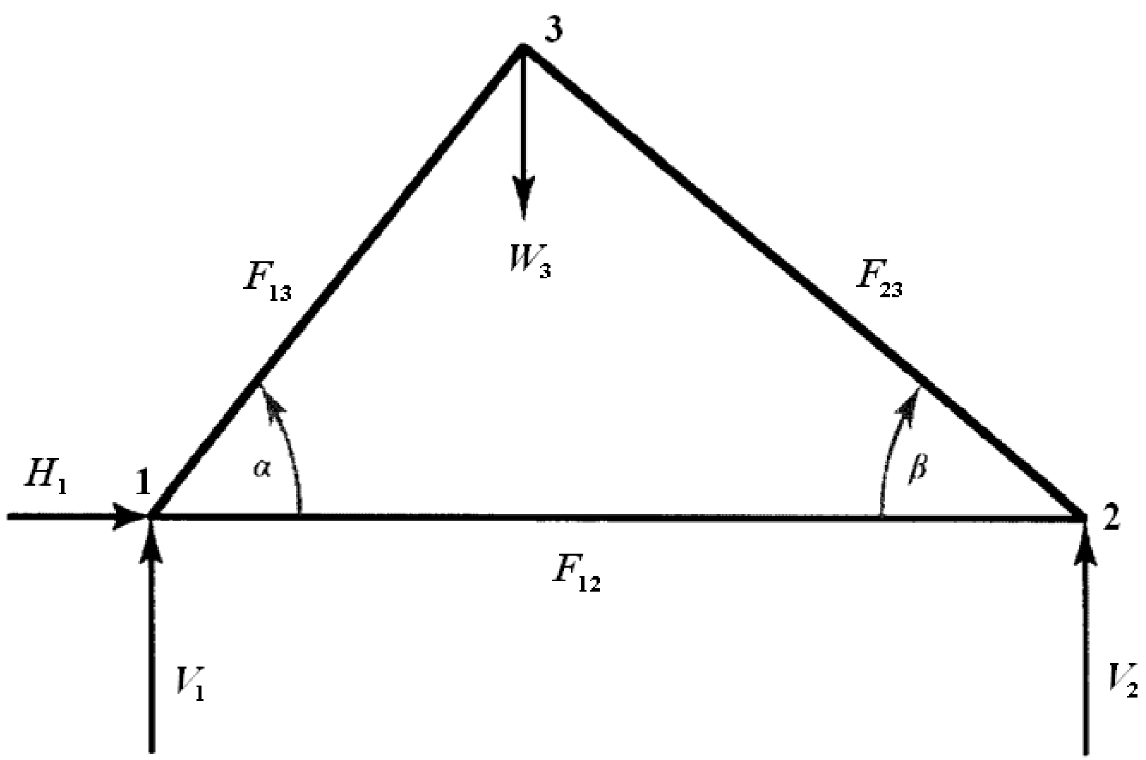
```
--------------------------------------------------
-------------- Gaussian Elimination -----------
Expansion Matrix

C =

   30   -20   -10     0
  -20    55   -10     0
  -10   -10    50   200

Solution:
x=[   3;   2;   5]
Run Time for Gauss Elimination Method =0.0160 (sec)
```


問題二：桁架的力學分析



問題二：桁架的力學分析

節點1

$$V_1 \qquad\qquad + F_{13}\sin(\alpha) = 0$$

$$H_1 + F_{12} + F_{13}\cos(\alpha) = 0$$

節點2

$$V_2 \qquad\qquad + F_{23}\sin(\beta) = 0$$

$$-F_{12} \qquad\qquad - F_{23}\cos(\beta) = 0$$

節點3

$$-F_{13}\sin(\alpha) - F_{23}\sin(\beta) = W_3$$

$$-F_{13}\cos(\alpha) + F_{23}\cos(\beta) = 0$$

問題二：桁架的力學分析

- 以矩陣AX=b來表示，求當W_3＝100及W_3＝75時的解

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & \sin(\alpha) & 0 \\ 0 & 1 & 0 & 1 & \cos(\alpha) & 0 \\ 0 & 0 & 1 & 0 & 0 & \sin(\beta) \\ 0 & 0 & 0 & -1 & 0 & -\cos(\beta) \\ 0 & 0 & 0 & 0 & -\sin(\alpha) & -\sin(\beta) \\ 0 & 0 & 0 & 0 & -\cos(\alpha) & \cos(\beta) \end{bmatrix}, b = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 100 & 75 \\ 0 & 0 \end{bmatrix}, \mathrm{x} = \begin{bmatrix} V_1 \\ H_1 \\ V_2 \\ F_{12} \\ F_{13} \\ F_{23} \end{bmatrix}$$

問題二：桁架的力學分析

- **主程式：**

```
clear all;
close all;

a=pi/6; b=pi/3;
A=[1 0 0  0   sin(a)    0
     0 1 0  1  cos(a)    0
     0 0 1  0      0      sin(b)
     0 0 0 -1      0      -cos(b)
     0 0 0  0 -sin(a)   -sin(b)
     0 0 0  0 -cos(a)   cos(b)];
b=[0    0
    0    0
    0    0
    0    0
  100 75
    0    0];

tic;
x=Gauss(A,b);
Gauss_Time=toc;
fprintf('Solution: \n');
x
fprintf('\nRun Time for Gauss Elimination Method =%6.4f (sec)\n\n',Gauss_Time);
```


問題二：桁架的力學分析

- **結果：**

```
--------------------------------------------------
-------------- Gaussian Elimination -----------
Expansion Matrix
C =
   1.0000         0         0         0    0.5000         0          0          0
        0    1.0000         0    1.0000    0.8660         0          0          0
        0         0    1.0000         0         0    0.8660          0          0
        0         0         0   -1.0000         0   -0.5000          0          0
        0         0         0         0   -0.5000   -0.8660   100.0000    75.0000
        0         0         0         0   -0.8660    0.5000          0          0

Solution:
x =
  25.0000   18.7500
        0         0
  75.0000   56.2500
  43.3013   32.4760
 -50.0000  -37.5000
 -86.6025  -64.9519

Run Time for Gauss Elimination Method =0.0320 (sec)
```

第一行為節點3承受100單位力的結果;
第二行則為第二種情況的結果。

基本高斯消去法之分析

- 在基本高斯消去法是假設一定可以找到所須的乘數，以將各行化簡為零。如果遇到樞軸元素為零時，可能會出現兩種情況，差別在於該樞軸行，在樞軸列之下是否有非零元素。
- 如果出現一個零樞軸元素(在*n* x *n*線性方程組中)，且該樞軸行在樞軸列之下全為零，則該方程組沒有唯一解。此方程組可能是不一致 (Inconsistent)或是贅餘(Redundant)。

基本高斯消去法之分析

- 如果在樞軸位置出現零，但該樞軸行在樞軸元素之下有一個非零元素，可以修改高斯消去法，將樞軸元素為零的列，與其下方的列對調。此一程序叫做(部份)樞軸變換(Pivoting)，此為下一節的主題。
- 如果存在有唯一解、如果計算為全真的、且在任何階段樞軸元素都不為零，高斯消去法可以求得解答。但是計算並非全真的，會遇到捨入造成的計算誤差。

基本高斯消去法之分析

- 考慮以下兩個變數的兩個方程式所成的方程組，並假設在高斯消去法的每一階段，所使用的計算都捨入到兩位數:

$$0.001x_1 + x_2 = 3$$

$$x_1 + 2x_2 = 5$$

基本高斯消去法之分析

- 依照基本高斯消去法的步驟進行，我們將第一式乘以-1000再加到第二式，可得(若計算為全真的)

$$-998x_2 = \text{-2995捨入後}$$

$$\text{-}1000x_2 = \text{-}3000$$

$$\therefore x_2 = 3, \text{則} x_1 = 0$$

- 很明顯，對第二式的解而言，這不是一個好的近似值。需改進高斯消去法，以避免出現此種困境。

不良條件(Ill-Condition)

- 所謂不良條件是指線性方程組或矩陣中的係數或等號右側之向量有微小變動時，就會造成其解答發生急劇變化稱之。

$$x_1+\frac{1}{2}x_2=\frac{3}{2}$$
$$\frac{1}{2}x_1+\frac{1}{3}x_2=\frac{5}{6}$$

其解為$x_1=x_2=1$

將等號右側稍加修改成

$$x_1+\frac{1}{2}x_2=\frac{3}{2}$$
$$\frac{1}{2}x_1+\frac{1}{3}x_2=1$$

其解為$x_1=0, x_2=3$

不良條件(Ill-Condition)

- 對等號右側的微幅變化極端敏感的情形，代表其係數矩陣是「病態條件的(ill-conditioned)」。由病態條件所造成的困難，無法經由修改高斯消去法而加以克服。
- 如果將矩陣的條件數(condition number)定義為該矩陣最大之特徵值與最小之特徵值的比值，則條件數大的矩陣就是病態矩陣。用MATLAB的內建函數cond可以求得矩陣的條件數。Cond(H)

列樞軸(Row Pivoting)變換

- 在高斯消去法的任一階段，若樞軸元素為零，則基本高斯消去法就不能用。
- 另外，當樞軸元素遠小於它要去消掉的元素，使得計算結果的不準確度相當高。將擴張矩陣中的某些列調換順序解決之，此方法稱為列樞軸變換。

例3.4 不良條件方程組

$0.001x_1 + x_2 = 3$

$x_1 + 2x_2 = 5$

使用基本高斯消去法，

再捨入後，其解為 $x_2 = 3, x_1 = 0$

列樞軸變換

$x_1 + 2x_2 = 5$

$0.001x_1 + x_2 = 3$

將第一式乘以 -0.001 再加到第二式

$0.998x_2 = 2.995$

在捨入後，其解為 $x_2 = 3, x_1 = -1$

例3.5 用樞軸變換高斯消去法

• 有一線性方程組：

$$\begin{aligned} 2x + \ \ 6y + 10z &= 0 \\ x + \ \ 3y + \ \ 3z &= 2 \\ 3x + 14y + 28z &= -8 \end{aligned}$$

$A\mathrm{x} = b$,其中

$$\mathrm{A} = \begin{bmatrix} 2 & 6 & 10 \\ 1 & 3 & 3 \\ 3 & 14 & 28 \end{bmatrix}, b = \begin{bmatrix} 0 \\ 2 \\ -8 \end{bmatrix}, \mathrm{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

例3.5 用樞軸變換高斯消去法

其擴張矩陣為

$$\left[\begin{array}{ccc|c} 2 & 6 & 10 & 0 \\ 1 & 3 & 3 & 2 \\ 3 & 14 & 28 & -8 \end{array}\right]$$

因為 $a_{31} > a_{11}$，將列 1 及 3 對調。

$$\left[\begin{array}{ccc|c} 3 & 14 & 28 & -8 \\ 1 & 3 & 3 & 2 \\ 2 & 6 & 10 & 0 \end{array}\right]$$

例3.5 用樞軸變換高斯消去法

轉換列 2 和 3 以得到

$$\left[\begin{array}{cccc|c} 3 & 14 & 28 & & -8 \\ 0 & -5/3 & -19/3 & & 14/3 \\ 0 & -10/3 & -26/3 & & 16/3 \end{array}\right]$$

因為 $|a_{33}|>|a_{22}|$，將列 2 和 3 對調。

$$\left[\begin{array}{ccc|c} 3 & 14 & 28 & -8 \\ 0 & -10/3 & -26/3 & 16/3 \\ 0 & -5/3 & -19/3 & 14/3 \end{array}\right]$$

例3.5 用樞軸變換高斯消去法

轉換列 3 以獲得

$$\left[\begin{array}{ccc|c} 3 & 14 & 28 & -8 \\ 0 & -10/3 & -26/3 & 16/3 \\ 0 & 0 & -2 & 2 \end{array}\right]$$

利用後向代換，

$$x_3 = \frac{a_{34}}{a_{33}} = \frac{2}{-2} \Rightarrow x_3 = -1$$

$$x_2 = \frac{a_{24} - a_{23}x_3}{a_{22}} = \frac{16/3 - (-26/3)(-1)}{-10/3} \Rightarrow x_2 = 1$$

$$x_1 = \frac{a_{14} - a_{12}x_2 - a_{13}x_3}{a_{11}} = \frac{-8 - 14(1) - 28(-1)}{3} \Rightarrow x_1 = 2$$

使用MATLAB程式

- 主程式：Main_GaussPivot.m
- 含樞軸變換之高斯消去法：GaussPivot.m
- 題目：

$$\begin{aligned} 2x + 6y + 10z &= 0 \\ x + 3y + 3z &= 2 \\ 3x + 14y + 28z &= -8 \end{aligned}$$

使用MATLAB程式

- 結果：

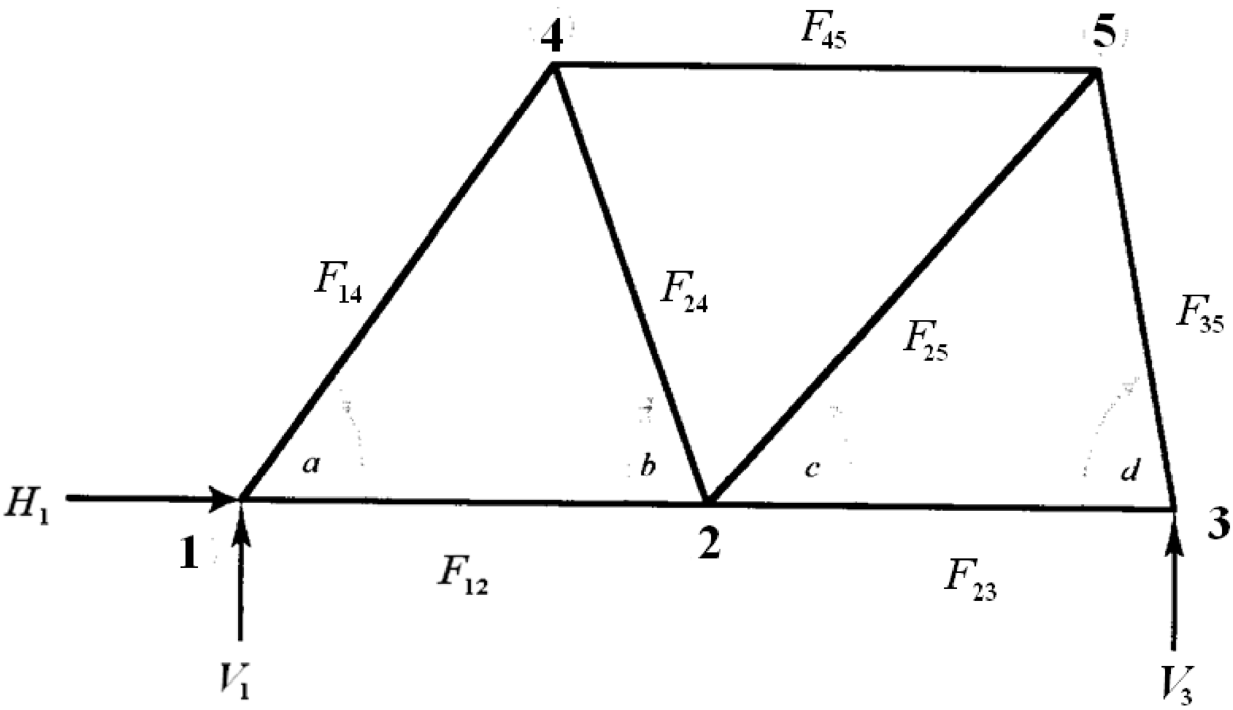
```
---------------------------------------------------
-------------- Gaussian Elimination with Pivoting-----------
Expansion Matrix

C =

    2     6    10     0
    1     3     3     2
    3    14    28    -8

Solution:
x=[   2;   1;  -1]
Run Time for Gauss Elimination with Pivoting =0.0160 (sec)
```

例題3.6 Simple Truss



例題3.6 Simple Truss

```
A = [  1    0    0    0   sin(a)   0      0        0        0       0
       0    1    0    1   cos(a)   0      0        0        0       0
       0    0    0    0    0       0    sin(b)   sin(c)     0       0
       0    0    0   -1    0       1   -cos(b)   cos(c)     0       0
       0    0    1    0    0       0      0        0      sin(d)    0
       0    0    0    0    0      -1      0        0     -cos(d)    0
       0    0    0    0  -sin(a)   0   -sin(b)     0        0       0
       0    0    0    0  -cos(a)   0    cos(b)     0        0       1
       0    0    0    0    0       0      0     -sin(c)  -sin(d)    0
       0    0    0    0    0       0      0     -cos(c)   cos(d)   -1   ]
x = [ V1   H1   V3   F12   F14   F23   F24   F25   F35   F45 ]'
```

假設此結構之重量 (100kg) 集中在節點 2，

```
r = [ 0;  0;  100;  0;  0;  0;  0;  0;  0;  0  ]
```

在角度爲 $a = b = c = d = \pi/4$ 時求各點受力，在定義矩陣 **A** 之前要先定義這些值。

使用MATLAB程式

- 主程式：Main_GaussPivot.m
- 含樞軸變換之高斯消去法：GaussPivot.m

```
clear all;
close all;

a=pi/4; b=a; c=a; d=a;
A=[1   0   0   0  sin(a)   0       0         0       0       0
   0   1   0   1  cos(a)   0       0         0       0       0
   0   0   0   0    0      0     sin(b)  sin(c)     0       0
   0   0   0  -1    0      1    -cos(b)  cos(c)     0       0
   0   0   1   0    0      0       0        0      sin(d)   0
   0   0   0   0    0     -1       0        0     -cos(d)   0
   0   0   0   0  -sin(a)  0    -sin(b)     0        0      0
   0   0   0   0  -cos(a)  0     cos(b)     0        0      1
   0   0   0   0    0      0       0     -sin(c)  -sin(d)   0
   0   0   0   0    0      0       0     -cos(c)   cos(d)  -1
   ];
r=[0;0;100;0;0;0;0;0;0;0];

tic;
x=GaussPivot(A,r);
Gauss_Time=toc;
disp('x=[V1 H1 V3 F12 F14 F23 F24 F25 F35 F45]');
fprintf('Solution: \n');
fprintf('x=[');
fprintf('\n  %7.4f',x);
fprintf(' ];');
fprintf('\nRun Time for Gauss Elimination with Pivoting =%6.4f (sec)\n\n',Gauss_Time);
```


使用MATLAB程式

- 結果：

```
x=[V1 H1 V3 F12 F14 F23 F24 F25 F35 F45]
Solution:
x=[
  50.0000
   0.0000
  50.0000
  50.0000
  -70.7107
  50.0000
  70.7107
  70.7107
  -70.7107
  -100.0000 ];
Run Time for Gauss Elimination with Pivoting =0.0470
(sec)
```

分析

- 對於不良條件的矩陣，在每一行中找出最大的值置換至可當作樞軸處，再進行高斯消去法。
- 另外，可使用尺度化(Scaling Strategy)方式
  - ➢ *一種可能的做法是將每一列分別乘以合適的10的冪次(如果右側項是用單獨的向量儲存，也要同時做尺度化)以使得係數矩陣之每一列中，絕對值最大的元素是介於0.1和1之間。用2的冪次來乘會更好，因為這樣可避免產生捨入誤差。*

分析

➢ *將A的每一列，除以該列中最大的元素，同時對b執行同樣的動作，不過這可能會造成額外的捨入誤差。*

高斯-喬丹消去法(Gauss-Jordan)

- 高斯-喬丹消去法
  - 1.先將對角線以下元素化簡為零(與標準高斯消去法一樣)；
  - 2.將對角線以上元素化簡(以最後一列為樞軸開始)至零(後向消去)。
  - 3.將各列尺度化，使得對角線元素為1。如此就將矩陣轉換成最簡列梯形型式(Reduced row echelon Form, RREF)。

$$\left(AI\right) \Rightarrow A^{-1}(AI) \Rightarrow (IA^{-1})$$

高斯-喬丹消去法(Gauss-Jordan)

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

$$(AI) = \begin{bmatrix} 2 & -1 & 0 & 1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 1 & 0 \\ 0 & -1 & 2 & 0 & 0 & 1 \end{bmatrix}$$

$$(IA^{-1}) = \begin{bmatrix} 1 & 0 & 0 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 1 & 0 & \frac{1}{2} & 1 & \frac{1}{2} \\ 0 & 0 & 1 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} \end{bmatrix}$$

高斯-喬丹消去法(Gauss-Jordan)

- 為了計算效率的考量，消去的過程分兩階段進行。在整個前向消去的過程都完成後，再進行後向消去是比較有效率的。
- 在以下高斯-喬丹消去法的函數中，在後向消去階段，只有可能會出現非零元素的行會被更新。

例3.5 用高斯-喬丹法求線性方程組

- 有一線性方程組：

$$\begin{aligned} 2x + 6y + 10z &= 0 \\ x + 3y + 3z &= 2 \\ 3x + 14y + 28z &= -8 \end{aligned}$$

$A\mathrm{x} = b$,其中

$$\mathrm{A} = \begin{bmatrix} 2 & 6 & 10 \\ 1 & 3 & 3 \\ 3 & 14 & 28 \end{bmatrix}, b = \begin{bmatrix} 0 \\ 2 \\ -8 \end{bmatrix}, \mathrm{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

例3.5： MATLAB程式

- **主程式：Main_GaussJordan.m**
- **演算程式：GaussJordan.m**

```
clear all;
close all;

A=[2 6 10;1 3 3;3 14 28];
b=[0; 2; -8];

tic;
x=GaussJordan(A,b);
Gauss_Time=toc;
fprintf('Solution: \n');
fprintf('x=[');
fprintf('\n  %7.4f',x’);
fprintf(' ];');
fprintf('\nRun Time for Gauss-Jordan with Pivoting =%6.4f (sec)\n\n',Gauss_Time);
```

例3.5： MATLAB程式

- **結果：**

```
C =

    2     6    10     0
    1     3     3     2
    3    14    28    -8

Solution:
x=[
   2.0000
   1.0000
  -1.0000 ];
Run Time for Gauss-Jordan with Pivoting =0.0470 (sec)
```

以高斯-喬丹消去法求逆矩陣

- 求下列矩陣之逆矩陣

$$A=\begin{bmatrix} 3 & -1 & 2 \\ 1 & 1 & 2 \\ 2 & -2 & -1 \end{bmatrix}$$

- 使用**AX=B**，而

$$B=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

MATLAB程式

- 主程式：**Main_GaussJordan_Ex2.m**
- 演算程式：**GaussJordan.m**

```
clear all;
close all;

A=[3 -1 2;1 1 2;2 -2 -1];
b=eye(3);

tic;
x=GaussJordan(A,b);
Gauss_Time=toc;
fprintf('Solution: \n');
fprintf('x=[');
fprintf('\n  %7.4f  %7.4f  %7.4f',x');
fprintf(' ];');
fprintf('\nRun Time for Gauss-Jordan with Pivoting =%6.4f (sec)\n\n',Gauss_Time);
```

MATLAB程式

- 結果：

```
---------------------------------------------------
-------------- Gaussian-Jordan Method-----------

C =

     3    -1     2     1     0     0
     1     1     2     0     1     0
     2    -2    -1     0     0     1

Solution:
x=[
  -0.7500   1.2500   1.0000
  -1.2500   1.7500   1.0000
   1.0000  -1.0000  -1.0000 ];
Run Time for Gauss-Jordan with Pivoting =0.0620 (sec)
```


高斯-喬丹消去法(Gauss-Jordan)

- 高斯-約丹消去法中納入樞軸變換(至少是在高斯消去法中所述的列樞軸變換)以獲得數值方法的穩定性。完整的樞軸變換包括列與行，但是這會增加此方法記錄的複雜度，而只能微幅增進其穩定性。

Homework 3

- 例題3.6之桁架，求
  - *1. 用手計算*
  - *2. 用高斯消去法計算*
  - *3. 用高斯-喬丹消去法計算*
  - *4. 比較上兩種數值方法*

三對角線方程組 (Tridiagonal Systems)

- 求解的線性方程組具有帶狀結構。一個三對角線方程組，其係數矩陣中只有對角線、下對角線(Subdiagonal)和上對角線(Superdiagonal)有非零元素。為應用此種結構的特性，我們使用修改的高斯消去法，在每一個消去步驟之後，我們將樞軸列尺度化，使得對角線元素為1。

例題3.8 用於三對角線方程組的尺度化高斯消去法

- 有一方程組

$$
\begin{aligned}
2x_1 + 2x_2 \qquad\qquad\qquad &= 4 \\
2x_1 + 4x_2 + 4x_3 \qquad &= 6 \\
x_2 + 3x_3 + 3x_4 &= 7 \\
2x_3 + 5x_4 &= 10
\end{aligned}
$$

首先，將第一式通除 a_{11} 以調整其尺度：

$$
\begin{aligned}
x_1 + x_2 \qquad\qquad\qquad &= 2 \\
2x_1 + 4x_2 + 4x_3 \qquad &= 6 \\
x_2 + 3x_3 + 3x_4 &= 7 \\
2x_3 + 5x_4 &= 10
\end{aligned}
$$

然後用第一式以消去第二式中的 x_1 項：

$$
\begin{aligned}
x_1 + x_2 \qquad\qquad\qquad &= 2 \\
2x_2 + 4x_3 \qquad &= 2 \\
x_2 + 3x_3 + 3x_4 &= 7 \\
2x_3 + 5x_4 &= 10
\end{aligned}
$$

將第二式尺度化以完成此步驟：

$$
\begin{aligned}
x_1 + x_2 \qquad\qquad\qquad &= 2 \\
x_2 + 2x_3 \qquad &= 1 \\
x_2 + 3x_3 + 3x_4 &= 7 \\
2x_3 + 5x_4 &= 10
\end{aligned}
$$

其次，用第二式以消去第三式中的 x_2 項：

$$\begin{aligned} x_1 + x_2 \quad\quad\quad &= 2 \\ x_2 + 2x_3 \quad &= 1 \\ x_3 + 3x_4 &= 6 \\ 2x_3 + 5x_4 &= 10 \end{aligned}$$

第三式無須尺度化，所以我們以第三式消去第四式中的 x_3 項：

$$\begin{aligned} x_1 + x_2 \quad\quad\quad &= 2 \\ x_2 + 2x_3 \quad &= 1 \\ x_3 + 3x_4 &= 6 \\ - x_4 &= -2 \end{aligned}$$

並將最後一式尺度化：

$$\begin{aligned} x_1 + x_2 \quad\quad\quad &= 2 \\ x_2 + 2x_3 \quad &= 1 \\ x_3 + 3x_4 &= 6 \\ x_4 &= 2 \end{aligned}$$

例題3.8 用於三對角線方程組的尺度化高斯消去法

現在用後向代換以獲得

$$\begin{aligned} x_4 &= 2 \\ x_3 &= 6-(3)(2) = 0 \\ x_2 &= 1-(2)(0) = 1 \\ x_1 &= 2-(1)(1) = 1 \end{aligned}$$

三對角線方程組(Tridiagonal Systems)

- 因為矩陣A的特殊形式，我們可以將儲存的需求由n^2減至3n，只儲存包含主對角線元素的向量d、包含對角線之上元素的向量a、以及包含對角線之下元素的向量b。留意到元素b_1和a_n都是零。右側項以向量r儲存。

三對角線方程組(Tridiagonal Systems)

- 三對角線方程組的通式：

$$
\begin{aligned}
&d_1x_1 + a_1x_2 && = r_1 \\
&b_2x_1 + d_2x_2 + a_2x_3 && = r_2 \\
&\qquad\ddots \qquad\qquad \ddots && \\
&\qquad\qquad b_{n-1}x_{n-2} + d_{n-1}x_{n-1} + a_{n-1}x_n && = r_{n-1} \\
&\qquad\qquad\qquad b_nx_{n-1} + d_nx_n && = r_n
\end{aligned}
$$

$$\begin{cases} d_1x_1 + a_1x_2 & = r_1 \\ b_2x_1 + d_2x_2 + a_2x_3 = r_2 \end{cases}$$

$$\begin{cases} x_1 + \dfrac{a_1}{d_1}x_2 & = \dfrac{r_1}{d_1} \\ b_2x_1 + d_2x_2 + a_2x_3 = r_2 \end{cases}$$

第一列$\times(-b_2)+$第二列

$$\begin{cases} x_1 + \dfrac{a_1}{d_1}x_2 & = \dfrac{r_1}{d_1} \\ (b_2 - b_2)x_1 + (d_2 - \dfrac{a_1}{d_1}b_2)x_2 + a_2x_3 = (r_2 - \dfrac{r_1}{d_1}b_2) \end{cases}$$

$$x_1 + \frac{a_1}{d_1}x_2 = \frac{r_1}{d_1} \Longrightarrow x_1 + (a_1*)x_2 = r_1*$$

$$\begin{cases} x_1 + (a_1*)x_2 & = r_1* \\ (b_2 - b_2)x_1 + (d_2 - a_1*b_2)x_2 + a_2x_3 = (r_2 - r_1*b_2) \end{cases}$$

$$\begin{cases} x_1 + (a_1*)x_2 & = r_1* \\ x_2 + \dfrac{a_2}{(d_2 - a_1*b_2)}x_3 = \dfrac{(r_2 - r_1*b_2)}{(d_2 - a_1*b_2)} \end{cases}$$

$$\boxed{a_2* = \frac{a_2}{(d_2 - a_1*b_2)}, r_2* = \frac{(r_2 - r_1*b_2)}{(d_2 - a_1*b_2)}}$$

Thomas演算法

- 步驟1：對第一式，產生新的元素a_1及r_1

$$a_1 = \frac{a_1}{d_1}, r_1 = \frac{r_1}{d_1}$$

- 步驟2：對i=2,3,…,n-1的每一個方程式

$$a_i = \frac{a_i}{d_i - b_i a_{i-1}}, r_i = \frac{r_i - b_i r_{i-1}}{d_i - b_i a_{i-1}}$$

Thomas演算法

- 步驟3：對最後一式

$$r_n = \frac{r_n - b_n r_{n-1}}{d_n - b_n a_{n-1}}$$

- 步驟4：以後向代換求解

$$x_n = r_n$$

$$x_i = r_i - a_i x_{i+1}, (i = n-1, n-2, \cdots, 2, 1)$$

例題3.9 用Thomas演算法解三對角線方程組

- 有一方程組

$$
\begin{aligned}
2x_1 + 2x_2 \qquad\qquad\quad &= 4 \\
2x_1 + 4x_2 + 4x_3 \qquad &= 6 \\
x_2 + 3x_3 + 3x_4 &= 7 \\
2x_3 + 5x_4 &= 10
\end{aligned}
$$

- 已知對角線d、對角線之上a、對角線之下b和右側項r等向量為

$$
\begin{aligned}
\mathbf{d} &= \begin{bmatrix} 2 & 4 & 3 & 5 \end{bmatrix} \\
\mathbf{a} &= \begin{bmatrix} 2 & 4 & 3 & 0 \end{bmatrix} \\
\mathbf{b} &= \begin{bmatrix} 0 & 2 & 1 & 2 \end{bmatrix} \\
\mathbf{r} &= \begin{bmatrix} 4 & 6 & 7 & 10 \end{bmatrix}
\end{aligned}
$$

例題3.9 用Thomas演算法解三對角線方程組

首先，產生新元素 a_1 和 r_1：

$$a_1 = \frac{a_1}{d_1} = \frac{2}{2} = 1, \qquad r_1 = \frac{r_1}{d_1} = \frac{4}{2} = 2$$

對第二式：

$$a_2 = \frac{a_2}{d_2 - b_2 a_1} = \frac{4}{4-(2)(1)} = 2; \quad r_2 = \frac{r_2 - b_2 r_1}{d_2 - b_2 a_1} = \frac{6-(2)(2)}{4-(2)(1)} = 1$$

對第三式：

$$a_3 = \frac{a_3}{d_3 - b_3 a_2} = \frac{3}{3-(1)(2)} = 3; \quad r_3 = \frac{r_3 - b_3 r_2}{d_3 - b_3 a_2} = \frac{7-(1)(1)}{3-(1)(2)} = 6$$

例題3.9 用Thomas演算法解三對角線方程組

對最後一式：

$$r_4 = \frac{r_4 - b_4 r_3}{d_4 - b_4 a_3} = \frac{10-(2)(6)}{5-(2)(3)} = 2$$

最後，用後向代換以獲得

$$x_4 = r_4 = 2; \qquad x_3 = r_3 - a_3 x_4 = 6-(3)(2) = 0;$$

$$x_2 = r_2 - a_2 x_3 = 1-(2)(0) = 1; \quad x_1 = r_1 - a_1 x_2 = 2-(1)(1) = 1$$

例題3.9：MATLAB程式

- **主程式：Main_Thomas.m**
- **演算程式：Thomas.m**

```
clear all;
close all;

d=[2 4 3 5]; a=[2 4 3 0]; b=[0 2 1 2]; r=[4 6 7 10];

tic;
x = Thomas(a, d, b, r);
Thomas_Time=toc;
fprintf('Solution: \n');
fprintf('x=[');
fprintf('\n  %7.4f',x');
fprintf(' ];');
fprintf('\nRun Time for Thomas Method =%6.4f (sec)\n\n',Thomas_Time);
```

例題3.9：MATLAB程式

- 結果：

```
---------------------------------------------------
-------------- Thomas Method-----------
Solution:
x=[
   1.0000
   1.0000
   0.0000
   2.0000 ];
Run Time for Thomas Method =0.0470 (sec)
```


討論

- 利用三對角線矩陣的特殊結構除大幅減低儲存的需求，Thomas演算法所須的計算量也遠小於一般的高斯消去法。
- Thomas演算法所須的乘法與除法次數為:
  - *對於第一式，需要2個除法。*
  - *對其後的(n-2)個方程式，每個需要2個乘法和2個除法。*
  - *對最後一式，需要2個乘法和1個除法。*
  - *消去過程共需5+4(n-2)。*
  - *在後向代換過程中，需要(n-1)個乘法。*
- 要使用Thomas演算法，必須對每一個i值都需滿足

$$d_1 \neq 0 \text{且} d_i - b_i a_{i-1} \neq 0$$

Gauss-Thomas法遇到樞軸元素為零的處理

- 考慮方程組

$$
\begin{aligned}
2x_1 - x_2 &= 1\\
-x_1 + 2x_2 - x_3 &= 0\\
-x_2 + \frac{2}{3}x_3 - x_4 &= -\frac{4}{3}\\
-x_3 + 2x_4 - x_5 &= 0\\
-x_4 + 2x_5 - x_6 &= 0\\
-x_5 + 2x_6 &= 1
\end{aligned}
$$

Gauss-Thomas法遇到樞軸元素為零的處理

求解的過程依正常方式進行，先將第一式尺度化後再用以消去第二式中的 x_1 項。在將第二式尺度化後用以消去第三式中的 x_2 項，我們得到

$$
\begin{aligned}
x_1 - \frac{1}{2}x_2 &= \frac{1}{2}\\
x_2 - \frac{2}{3}x_3 &= \frac{1}{3}\\
- x_4 &= -1\\
-x_3 + 2x_4 - x_5 &= 0\\
-x_4 + 2x_5 - x_6 &= 0\\
-x_5 + 2x_6 &= 1
\end{aligned}
$$

Gauss-Thomas法遇到樞軸元素為零的處理

但現在我們無法將第三式尺度化，使其對角線元素爲 1，因爲 x_3 的係數是 0。但第三列中還有非零的係數(變數 x_4)，我們可以解出該變數，然後繼續進行。因此我們用第三式解 x_4 可得 x_4=1。現在跳過第四式，將已求得之 x_4 的值代入第五式。繼續進行消去過程，利用第五式以消去最後一式中的 x_5：

$$
\begin{array}{lll}
x_1 - \frac{1}{2}x_2 & = \frac{1}{2} & \\
x_2 - \frac{2}{3}x_3 & = \frac{1}{3} & \\
-\ x_4 & = -1 & (\text{解}) \\
-x_3 + 2x_4 - x_5 & = 0 & (\text{跳過}) \\
2x_5 - x_6 & = 1 & (x_4 = 1) \\
-x_5 + 2x_6 & = 1 &
\end{array}
$$

Gauss-Thomas法遇到樞軸元素為零的處理

將第五式尺度化，用它消去最後一式中的 x_5，並將最後一式尺度化：

$$
\begin{array}{ll}
x_1 - \frac{1}{2}x_2 & = \frac{1}{2} \\
x_2 - \frac{2}{3}x_3 & = \frac{1}{3} \\
-\ x_4 & = -1 \\
-x_3 + 2x_4 - x_5 & = 0 \\
x_5 - \frac{1}{2}x_6 & = \frac{1}{2} \\
x_6 & = 1
\end{array}
$$

Gauss-Thomas法遇到樞軸元素為零的處理

用後向代換求解

$$x_6 = 1$$

$$x_5 = \frac{1}{2} + \frac{1}{2} x_6 = 1$$

$$x_4 = 1 \quad \text{(之前已求得)}$$

$$x_3 = 2x_4 - x_5 = 1 \quad \text{(之前略)}$$

$$x_2 = \frac{1}{3} + \frac{2}{3} x_3 = 1$$

$$x_1 = \frac{1}{2} + \frac{1}{2} x_2 = 1$$

MATLAB所用的方法

- **在MATLAB中有兩種除法符號**
  - *前斜線(/)：A/B相當於B*inv(A)*
  - *倒斜線(\\)：反矩陣，A\\B相當於inv(A)*B*

過定及欠定方程組 (Over-and Under-determined Systems)

- 如果未知數的數目多於方程式的數目，則該方程組為欠定，如果未知數的數目小於方程式的數目，則方程組為過定。
- 兩種情形下，重要的都是，有多少方程式是線性獨立的。

最簡列梯形型式 (Reduced Row Echelon Form)

- MATLAB的函數rref可將一個矩陣轉換成最簡列梯形型式。
- 指令**R=rref(A)**會用高斯-喬丹消去法加部份樞軸變換，以求得矩陣A的最簡列梯形型式。它會用內定的容許誤差來檢測是否有可忽略不計的行元素。
- 若以**[R,j]=rref(A)**的方式呼叫函數，同時傳回向量j，它可使r=length(j)是此演算法理想的A之秩(Rank)。向量j的元素，指出線性方程組Ax=b中，那些變數為樞軸變數。

MATLAB程式

```
A=[1 1 -3 5
     3 -2 1 0
     4 -2 0 2];
[R,j]=rref(A)
R =
    1     0    -1     2
    0     1    -2     3
    0     0     0     0
j =
    1     2
```


最簡列梯形型式

- 用[R,j]=rref(A,tol)的方式呼叫函數，會讓函數使用所給的容許誤差來檢測矩陣的秩數。
- 捨入誤差可能會使得此演算法所求出的矩陣秩數，和使用函數rank、orth及null所求得的不一樣。
- 可以利用MATLAB的函數inv求出逆矩陣，來求解線性方程組Ax=b；但是通常，乘上逆矩陣並非解線性方程組最好的方法。

矩陣的範數(Norm)

- 矩陣的範數是用來度量矩陣大小(Magnitude)之數。
- 範數是對函數、向量和矩陣定義的一種度量形式。任何物件的範數值都是一個非負實數。使用範數可以測量兩個函數、向量或矩陣之間的距離。向量範數是度量向量長度的一種定義形式。
- 範數有多種定義形式，只要滿足下面的條件即可定義為一個範數。同一向量，採用不同的範數定義，可得到不同的範數值。

矩陣範數之性質

$$\|A\| > 0 \text{ and } \|A\| = 0 \text{ if and only if } A = 0$$

$$\|kA\| = |k| \cdot \|A\|$$

$$\|A + B\| \le \|A\| + \|B\|$$

$$\|AB\| \le \|A\| \cdot \|B\|$$

矩陣的條件數(Condition Number of a Matrix)

- 一個非奇異矩陣A的條件數，是用該矩陣的範數(norm)$\|A\|$及其逆矩陣的範數$\|A^{-1}\|$來定義的。因為一個矩陣有幾種不同的範數，所以也有幾種不同的條件數。
- 某些矩陣範數是導自(或從屬於)向量範數。即

$$A\text{的條件數} = \|A\| \cdot \|A^{-1}\|$$

矩陣的條件 (Condition of a Matrix)

- 最常用的三種向量範數為

$$\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n| \qquad \text{(Manhattan 或 taxi-cab 範數)}$$

$$\|x\|_2 = (|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2)^{1/2} \qquad \text{(歐幾里得範數)}$$

$$\|x\|_\infty = \max |x_i| \qquad \text{(Chebyshev 範數)}$$

一般而言，若 p 是 ≥ 1 的實數，則 p-norm 爲

$$\|x\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{1/p}$$

對一個從屬於向量範數的矩陣範數，其定義爲

$$\|\mathbf{A}\| = \sup \|\mathbf{A}x\| / \|x\|$$

在此 sup 是對所有非零向量來取。

矩陣的條件 (Condition of a Matrix)

$$\|\mathbf{A}\|_1 = \quad 大$$

$$= \max_j (|a_{1j}| + |a_{2j}| + \cdots + |a_{nj}|) \quad = \max_j \sum_{i=1}^{n} |a_{ij}|$$

$$\|\mathbf{A}\|_\infty = \quad 大$$

$$= \max_i (|a_{i1}| + |a_{i2}| + \cdots + |a_{in}|) \quad = \max_i \sum_{j=1}^{n} |a_{ij}|$$

$$\|\mathbf{A}\|_2 = (\mathbf{A}^H \mathbf{A})^{1/2} \text{的最大特徵值}$$

又稱爲頻譜範數 (spectral norm)

$\mathbf{A}^H$是 **A** 之轉置的共軛複數矩陣。$\mathbf{A}^H\mathbf{A}$ 具有非負的特徵值；這些特徵值的平方根爲 **A** 的奇異值。

矩陣的條件 (Condition of a Matrix)

- 有一些很重要的矩陣範數並不從屬於任何向量範數。最重要的一個就是歐幾里得(或稱Schur或Frobenius)範數。

$$\|\mathbf{A}\|_E = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}^2 \right)^{1/2}$$

矩陣的向量範數

- 求範數的 MATLAB指令
  - ***norm(A,p), 其中***

P值	意　義
1	1-norm的範數，(矩陣的列範數)，相當於max(sum(abs(A))
2	最大的奇異值(特徵值)
‘fro’	Frobenius 範數又稱歐幾里得範數，相當於sqrt(sum(diag(A'*A)))
inf	無限大範數，(矩陣的行範數)，相當於max(sum(abs(A‘)))

  - ***norm(x) is the Euclidean length of a vector x.***
  - ***norm(x)/sqrt(n): the root-mean-square (RMS)(均方根) value of an n-element vector x.***

矩陣的條件數
(Condition Number of a Matrix)

- 矩陣條件數的定義為

$$cond(A) = \|A\| \cdot \|A^{-1}\|$$

- 求矩陣條件數的 MATLAB指令
  - ***cond(A,p), 其中***

P值	意　義
1	1-norm的矩陣條件數(Manhattan或taxi-cab向量範數)
2	2-norm的矩陣條件數(Euclidean 歐幾里得向量範數)
‘fro’	Frobenius norm的矩陣條件數
inf	無限大範數的矩陣條件數(Chebyshev向量範數)

例題： 求矩陣範數與條件數

$$A=\begin{bmatrix} 5 & -4 & -7 \\ -4 & 2 & -4 \\ -7 & -4 & 5 \end{bmatrix}$$，其範數為

$$\|A\|_e = \text{Euclidean norm} = 14.69$$

$$\|A\|_\infty = 16$$

$$\|A\|_1 = 16$$

$$\|A\|_2 = \text{Spectral norm} = 12$$

例題： 求矩陣範數與條件數

- 求矩陣範數的MATLAB指令

```
norm(A,'fro') = 14.6969
norm(A,inf) = 16
norm(A,1) = 16
norm(A,2) =  12
```

- 求矩陣條件數的MATLAB指令

```
cond(A,1) = 4.4444
cond(A,2) = 2
cond(A,'fro') = 3.6742
cond(A,inf) = 4.4444
```

Homework 4

- 使用Gauss-Thomas法程式解下列題目：
  - ➢ *習題A3.7*
  - ➢ *習題A3.8*
  - ➢ *習題A3.9*