線性代數第7章
線性系統之應用

陳擎文老師

以下介紹課本的
數學定理，公式，範例

1. 分支流量分析

電線電流，管線內水流，石油流動，交通流量，
資金流動

2. 管線內水流分析

建立流量聯立方程式

總流量in =總流量out

每個節點流量in =節點流量out

線性系統之應用1：流量分析

- 1.分支流量分析：包括：電線電流，管線內水流，石油流動，交通流量，資金流動
- 2. 分支連結的點稱為節點（nodes）或接點（junctions），「流」在此位置點進行分流。
- (page. 69)

範例 1：用線性系統進行網路分析

- 求在每一個分支的流動分量？　　　(page. 70)

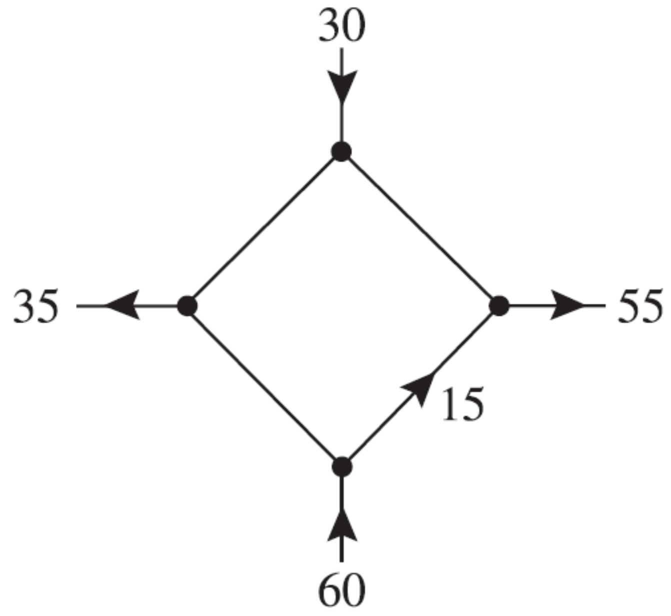


圖 1.9.1

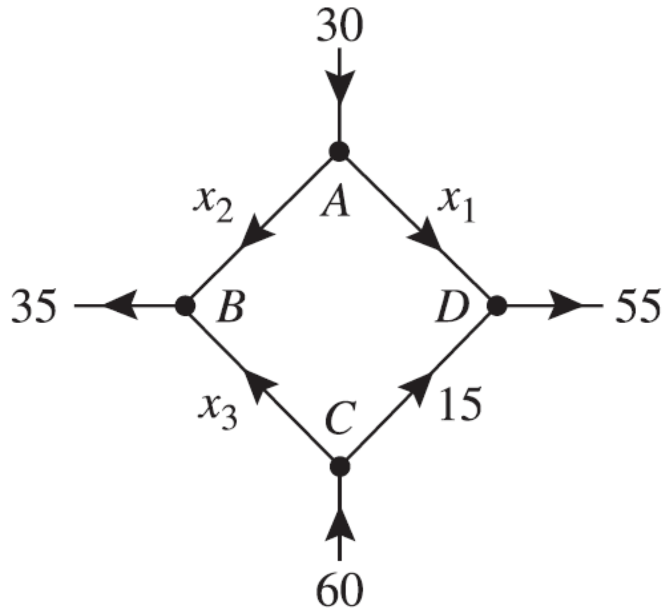
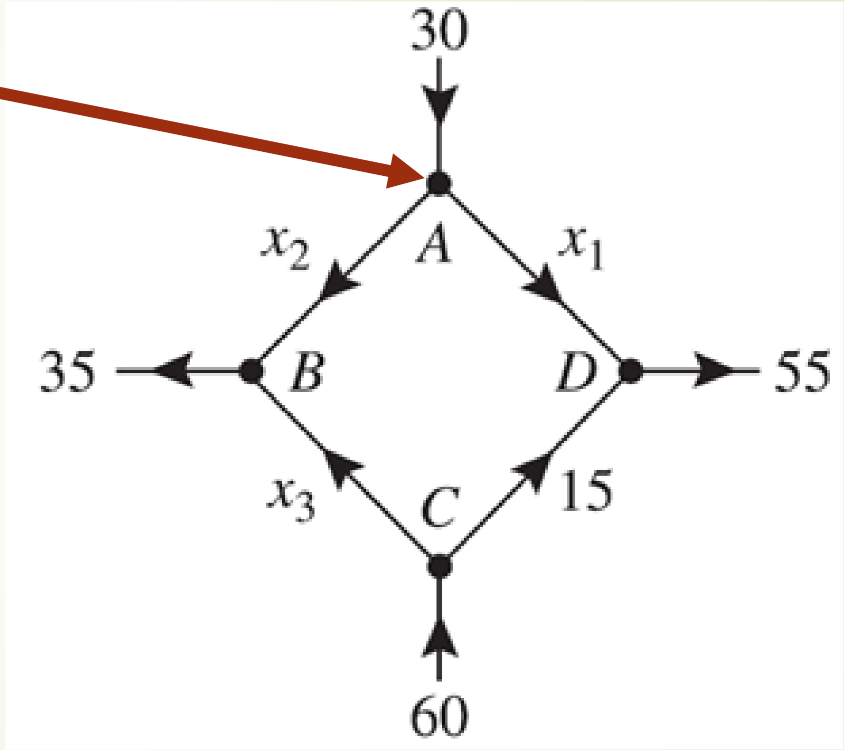


圖 1.9.2

範例 1：用線性系統進行網路分析

- 在每個節點看流量守恆關係式　　(page. 70)
  - 節點A：x1+x2=30
  - 節點B：x2+x3=35
  - 節點C：x3+15=60
  - 節點D：x1+15=55



- 線性系統擴展矩陣= $\begin{bmatrix} 1 & 1 & 0 & 30 \\ 0 & 1 & 1 & 35 \\ 0 & 0 & 1 & 45 \\ 1 & 0 & 0 & 40 \end{bmatrix}$
- 只能用高斯喬丹消去法，因為A不是nxn矩陣
- x1=40，x3=45 => x2=-10 =>x2的流量箭頭要向外畫

解聯立方程式的重點摘要-2

- 3. 計算聯立方程式的方法有三種：
  - (1). 高斯消去法
  - (2). Cramer’ s rule克拉瑪法則

    A必須為nxn矩陣
  - (3). 反矩陣法

    A必須為nxn矩陣

3. 交通流量分析

建立流量聯立方程式

總流量in =總流量out

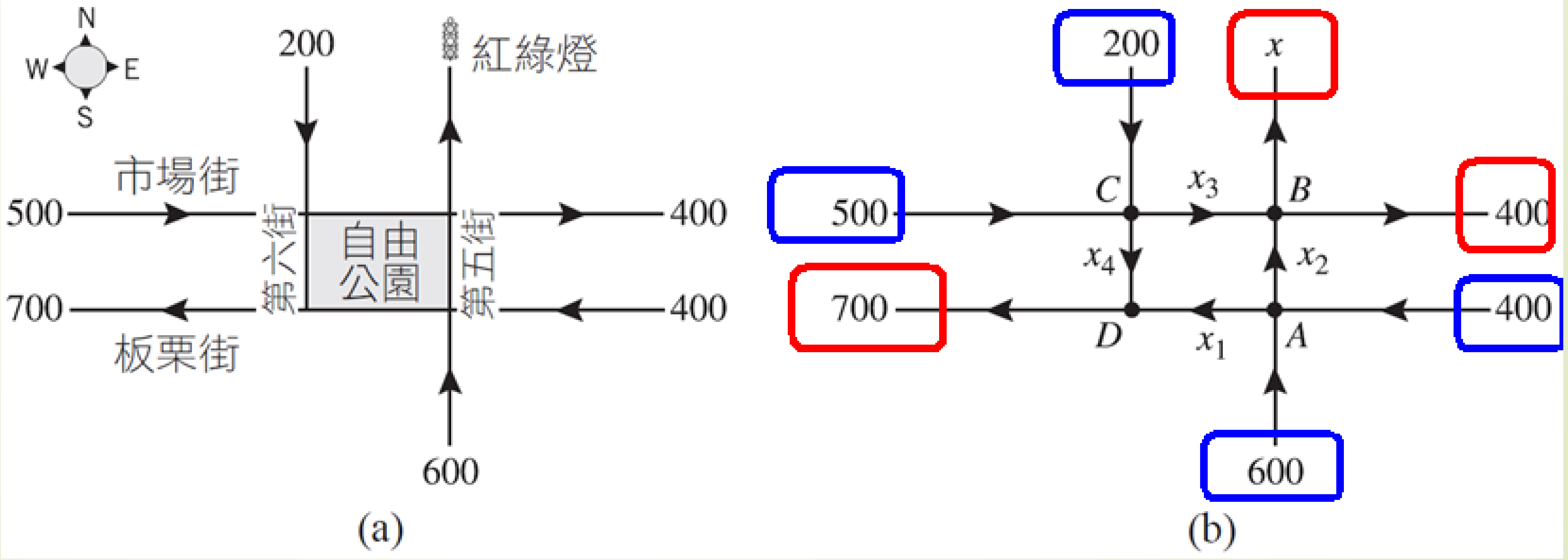
每個節點流量in =節點流量out

範例2：交通模式設計

要求要在第五街北出口進行電腦化紅綠燈 (page. 71)

示意圖是每小時流入和流出的平均汽車數量

1. 紅綠燈應每小時讓多少車通過，才能夠確保汽車流入和流出的車輛是一樣的？
2. 若流入和流出數量達到平衡，計算各街道每小時的平均汽車數？

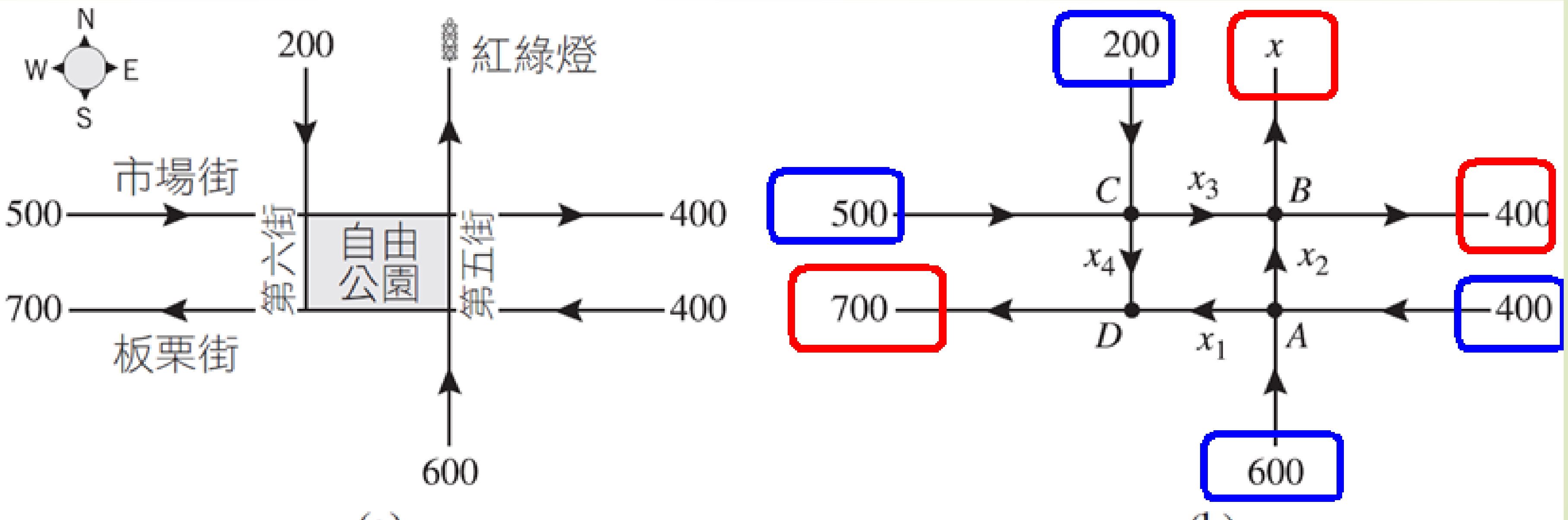


範例2：交通模式設計

1. 紅綠燈應每小時讓多少車通過，才能夠確保汽車流入和流出的車輛是一樣的？ (page. 71)

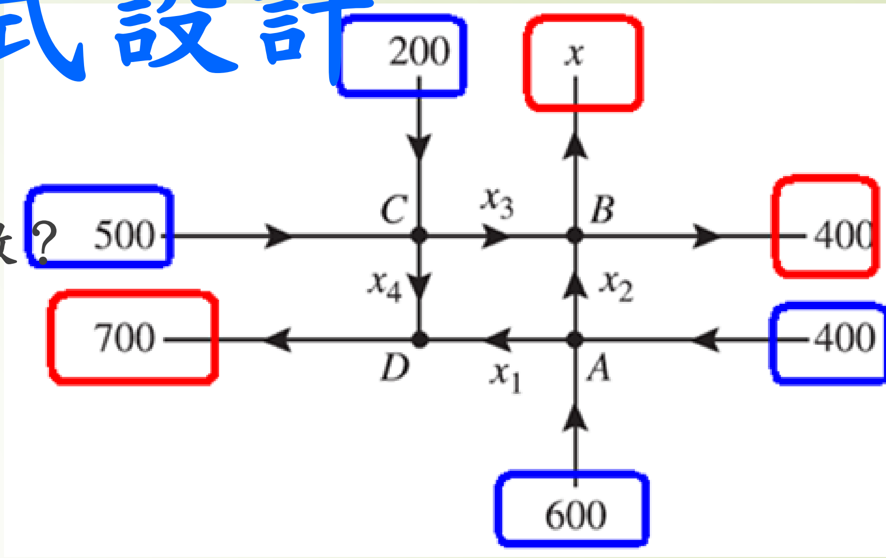
- 流出=700+400+x
- 流入=200+500+600+400
- x=1700−1100

  =600



(a) (b)

範例2：交通模式設計

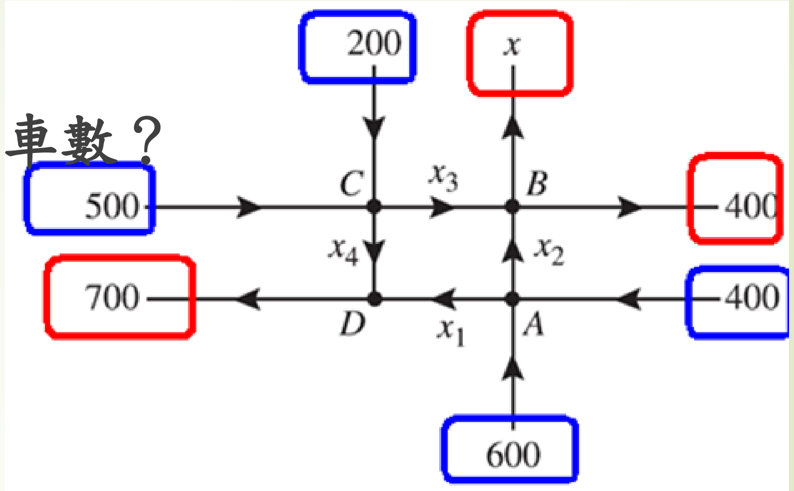


2. 若流入和流出數量達到平衡，計算各街道每小時的平均汽車數？

- 節點A：x1+x2=600+400=1000　　(page. 71)
- 節點B：x2+x3=600+400=1000
- 節點C：x3+x4=500+200=700
- 節點D：x1+x4=700

- 是否有解=det(A)= $\det\left(\begin{bmatrix}1&1&0&0\\0&1&1&0\\0&0&1&1\\1&0&0&1\end{bmatrix}\right)=\det\left(=\begin{bmatrix}1&1&0\\0&1&1\\0&0&1\end{bmatrix}\right)-\det\left(\begin{bmatrix}1&0&0\\1&1&0\\0&1&1\end{bmatrix}\right)$
- =(1)-(1)=0，**可能無解，可能無限多解，所以只能用高斯喬丹消去法**
- 線性系統擴展矩陣= $\begin{bmatrix}1&1&0&0&1000\\0&1&1&0&1000\\0&0&1&1&700\\1&0&0&1&700\end{bmatrix}=\begin{bmatrix}1&1&0&0&1000\\0&1&1&0&1000\\0&0&1&1&700\\0&-1&0&1&-300\end{bmatrix}=\begin{bmatrix}1&1&0&0&1000\\0&1&1&0&1000\\0&0&1&1&700\\0&0&1&1&700\end{bmatrix}$
- $=\begin{bmatrix}1&1&0&0&1000\\0&1&1&0&1000\\0&0&1&1&700\\0&0&0&0&0\end{bmatrix}$ => AB有一列零列=>無限多組解 => x4=t, x3=700-t, x2=300+t, x1=700-t

範例2：交通模式設計

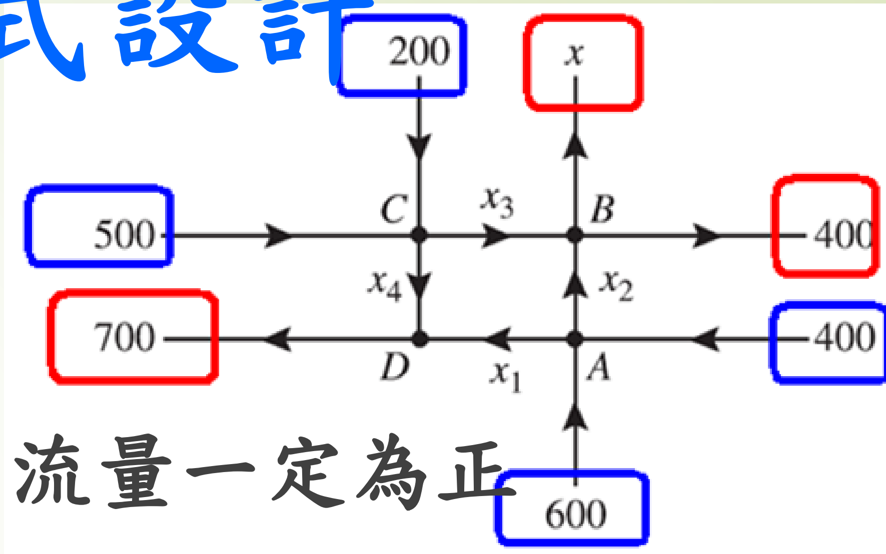


2. 若流入和流出數量達到平衡，計算各街道每小時的平均汽車數？

- 節點A：x1+x2=600+400=1000
- 節點B：x2+x3=600+400=1000
- 節點C：x3+x4=500+200=700
- 節點D：x1+x4=700

- 是否有解=det(A)=$det\left(\begin{bmatrix}1&1&0&0\\0&1&1&0\\0&0&1&1\\1&0&0&1\end{bmatrix}\right)$=det$\left(\begin{bmatrix}1&1&0\\0&1&1\\0&0&1\end{bmatrix}\right)$−det$\left(\begin{bmatrix}1&0&0\\1&1&0\\0&1&1\end{bmatrix}\right)$=1−1=0
- 可能無解，或無限多解
- $\det(A|B) = det\left(\begin{bmatrix}1&0&0&1000\\1&1&0&1000\\0&1&1&700\\0&0&1&700\end{bmatrix}\right)$ =det(=$\begin{bmatrix}1&0&1000\\1&1&700\\0&1&700\end{bmatrix}$)−det($\begin{bmatrix}0&0&1000\\1&1&700\\0&1&700\end{bmatrix}$)=(700+1000−700)−(1000)=0
- Det(A)=0 且 det(A|B)=0，所以無限多組解

範例2：交通模式設計



- 流量x1, x2, x3, x4的進一步限制？
- x4=t, x3=700-t, x2=300+t, x1=700-t
- （1）假設街道都是單向的，所以平均流量一定為正
  - 700-t >0 => t>700
  - t >0
  - 0 < t < 700
  - 0 < x4 < 700
  - 300 < x2 < 1000

$$0 \le x_1 \le 700, \quad 300 \le x_2 \le 1000, \quad 0 \le x_3 \le 700, \quad 0 \le x_4 \le 700$$

範例2：Python程式碼

```
from sympy import *
x1,x2,x3,x4 = symbols('x1 x2 x3 x4')
#注意：M矩陣的最右邊=y=轉換後座標
A = Matrix([
    [1,1,0,0,1000],
    [0,1,1,0,1000],
    [0,0,1,1,700],
    [1,0,0,1,700]
    ])
ans = solve_linear_system(A, x1, x2, x3, x4)
print('X=', ans)
```

```
X= {x3: 700 - x4, x2: x4 + 300, x1: 700 - x4}
```

重要

判別是否有唯一解的方法有兩種

- (1). 方法1：用增廣矩陣，將之化簡：簡化列梯形，然後評估每個變數是否存在？
  - 缺點：速度慢，高斯消去法，計算到最後才知道
  - 優點：可以判別是無解，還是無限多組解
- (2). 方法2：用原始矩陣A的行列式det(A)判別
  - 解聯立方程式之前要先判斷是否有唯一解=> det(A)≠
  - 若det(A)≠0，則系統有唯一解
  - 若det(A)=0，則系統可能無解，可能無限多組解
    - 若det(A)=0，且det(A|B)=0，則無限多組解
    - 若det(A)=0，且det(A|B)≠0，則無解

(a) $\left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ \boxed{0 \quad 0 \quad 0} & & & \boxed{1} \end{array}\right]$ 無解

(b) $\left[\begin{array}{cccc} 1 & 0 & 3 & -1 \\ 0 & 1 & -4 & 2 \\ 0 & 0 & 0 & 0 \end{array}\right]$ 無限多組解

範例2:Python 程式碼

```
import numpy as np
A = np.array([
    [1,1,0,0],
    [0,1,1,0],
    [0,0,1,1],
    [1,0,0,1]
    ])
Y = np.array([
    [1000],[1000],[700],[700]
    ])
AY = np.array([
    [1,0,0,1000],
    [1,1,0,1000],
    [0,1,1,700],
    [0,0,1,700]
    ])
```

```
A_det = np.linalg.det(A)
AY_det = np.linalg.det(AY)
if A_det !=0:
    X = np.linalg.solve(A, Y)
    print(’方程式有唯一解’, X)
elif A_det ==0 and AY_det==0:
    print(’方程式，無限多解’)
elif A_det ==0 and AY_det !=0:
    print(’方程式，無解’)
```

方程式，無限多解

4. 電路電流分析

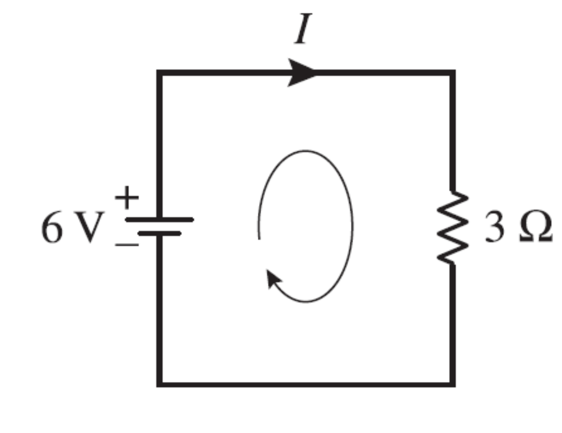
建立電流聯立方程式

每節點電流，電流in =電流out

每個封閉迴路，電壓升 =電壓降

範例3：一個封閉迴路的電路

- 試求圖 1.9.8 中電路的電流 I。
- （page. 74）

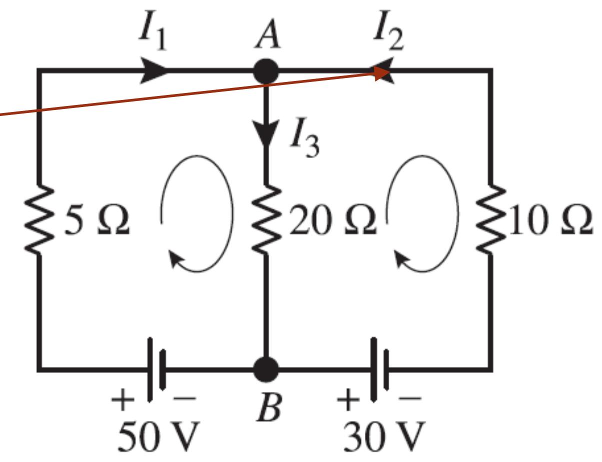


➲ 圖 1.9.8

- 電池會使電壓上升6伏特，以電阻會產
- 克希荷夫電壓定律：E = IR （電位差=電流*電阻）
- 電壓升 =電壓降
  - 電子流進電池（電壓升）
  - 電子流出電池（電壓降）
- 電壓升 =電壓降
- $6 = 3I$

範例 4：三個封閉迴路的電路

- 試求圖1.9.9中電路的電流 I_1, I_2 和 I_3。
- 計算電路電流方程式，必須用兩個定律
- (1). 克希荷夫電流定律：
  - 每個節點的電流，電流Iin =電流Iout
  - I1 +I2 = I3
- (2). 克希荷夫電壓定律:
  - 每個封閉迴路，電壓升 =電壓降
- (page. 74)



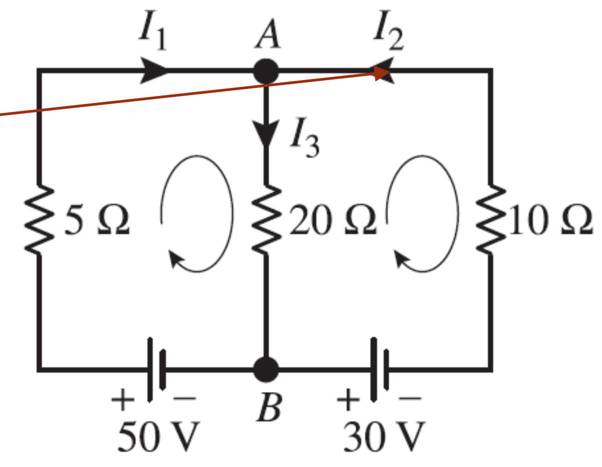
➲ 圖 1.9.9

節點	流入電流		流出電流
A	$I_1 + I_2$	=	I_3
B	I_3	=	$I_1 + I_2$

範例 4：三個封閉迴路的電路

- 試求圖1.9.9中電路的電流I_1, I_2 和I_3。
- 電流預訂方向：向右壓降，向左壓升
- 1. 看節點

節點	流入電流		流出電流
A	$I_1 + I_2$	$=$	I_3
B	I_3	$=$	$I_1 + I_2$



➲ 圖 1.9.9

- 2. 對三個封閉迴路，求電壓的升降如下
  - 電子流進電池（電壓升）
  - 電子流出電池（電壓降）

	電壓升	電壓降
左側內部迴路	50	$5I_1 + 20I_3$
右側內部迴路	$30 + 10I_2 + 20I_3$	0

（page. 74）

- 線性方程式

$$\begin{aligned} I_1 + I_2 - I_3 &= 0 \\ 5I_1 + 20I_3 &= 50 \\ 10I_2 + 20I_3 &= -30 \end{aligned}$$

- 因為A是nxn矩陣,
- 可用反矩陣法，用行列式法

解聯立方程式的重點摘要-2

- 3. 計算聯立方程式的方法有三種：
  - (1). 高斯消去法
  - (2). Cramer’s rule克拉瑪法則

    A必須為nxn矩陣
  - (3). 反矩陣法

    A必須為nxn矩陣

範例 4：三個封閉迴路的電路

- 線性方程式

$$\begin{aligned} I_1 + \quad I_2 - \quad I_3 &= 0 \\ 5I_1 \qquad\qquad + 20I_3 &= 50 \\ 10I_2 + 20I_3 &= -30 \end{aligned}$$

- $\begin{bmatrix} 1 & 1 & -1 & 0 \\ 5 & 0 & 20 & 50 \\ 0 & 10 & 20 & -30 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & 0 \\ 0 & -5 & 25 & 50 \\ 0 & 10 & 20 & -30 \end{bmatrix}$
- $= \begin{bmatrix} 1 & 1 & -1 & 0 \\ 0 & -1 & 5 & 10 \\ 0 & 1 & 2 & -3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & 0 \\ 0 & -1 & 5 & 10 \\ 0 & 0 & 7 & 7 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & 0 \\ 0 & -1 & 5 & 10 \\ 0 & 0 & 1 & 1 \end{bmatrix}$
- $= \begin{bmatrix} 1 & 1 & -1 & 0 \\ 0 & -1 & 0 & 5 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & -1 & 0 & 5 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 6 \\ 0 & -1 & 0 & 5 \\ 0 & 0 & 1 & 1 \end{bmatrix}$
- I1=6, I2=-5, I3=1

（page. 74）

範例 4:Python程式碼

```
from sympy import *
x1,x2,x3 = symbols('x1 x2 x3')
#注意：M矩陣的最右邊=y=轉換後座標
A = Matrix([
    [1,1,-1,0],
    [5,0,20,50],
    [0,10,20,-30]
    ])
ans = solve_linear_system(A, x1, x2, x3)
print('X=', ans)
```

```
X= {x1: 6, x2: -5, x3: 1}
```

5. 化學平衡式分析

建立化學元素聯立方程式

某原子元素左 =某原子元素右

範例：算甲烷燃燒反應式係數

- 甲烷燃燒反應式 $CH_4 + O_2 \rightarrow CO_2 + H_2O$

- 算甲烷燃燒反應式係數 $x_1\,(CH_4) + x_2\,(O_2) \rightarrow x_3\,(CO_2) + x_4\,(H_2O)$

- 必須找到正整數 x1, x2, x3, x4=？

（page. 76）

範例：算甲烷燃燒反應式係數

- 算甲烷燃燒反應式係數

$$x_1\,(CH_4) + x_2\,(O_2) \rightarrow x_3\,(CO_2) + x_4\,(H_2O)$$

左 = 右

碳C： x1 = x3

氫H：4x1 = 2x4

氧O：2x2 = 2x3 + x4

（page. 76）

- 解聯立方程式

$$\begin{aligned} x_1 \quad\quad - x_3 \quad\quad &= 0 \\ 4x_1 \quad\quad\quad\quad - 2x_4 &= 0 \\ 2x_2 - 2x_3 - x_4 &= 0 \end{aligned}$$

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

- 因為A不是nxn矩陣，只能用高斯喬丹消去法

範例：算甲烷燃燒反應式係數

- 算甲烷燃燒反應式係數

$$x_1\,(CH_4) + x_2\,(O_2) \rightarrow x_3\,(CO_2) + x_4\,(H_2O)$$

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

- 高斯喬丹法

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

- 無限多解

$$x_1 = t/2, \quad x_2 = t, \quad x_3 = t/2, \quad x_4 = t$$

- 最小整數發生在t=2
- X1=1, x2=2, x3=1, x4=2

範例5：算反應式係數

- 反應式

HCl	+	Na_3PO_4	$\longrightarrow$	H_3PO_4	+	$NaCl$
[鹽酸]	+	[磷酸鈉]	$\longrightarrow$	[磷酸]	+	[氯化鈉]

- 算反應式係數

令 x_1, x_2, x_3, x_4 為正整數，平衡的化學方程式為

$$x_1\,(HCl) + x_2\,(Na_3PO_4) \longrightarrow x_3\,(H_3PO_4) + x_4\,(NaCl)$$

- 找正整數 x1, x2, x3, x4=？

（page. 77）

範例5：算反應式係數

- 左=右　　　(page. 77)

$1x_1 = 3x_3$	氫 (H)
$1x_1 = 1x_4$	氯 (Cl)
$3x_2 = 1x_4$	鈉 (Na)
$1x_2 = 1x_3$	磷 (P)
$4x_2 = 4x_3$	氧 (O)

$$\begin{aligned} x_1 \quad\quad - 3x_3 \quad &= 0 \\ x_1 \quad\quad\quad\quad - x_4 &= 0 \\ 3x_2 \quad\quad - x_4 &= 0 \\ x_2 - x_3 \quad &= 0 \\ 4x_2 - 4x_3 \quad &= 0 \end{aligned}$$

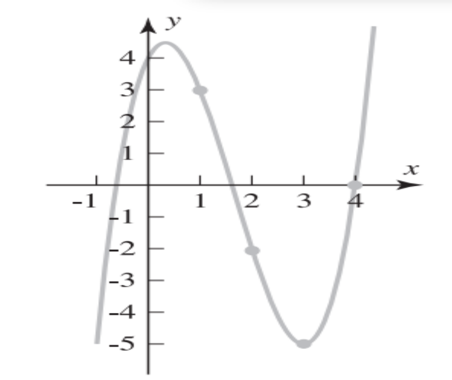
→

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & -\frac{1}{3} & 0 \\ 0 & 0 & 1 & -\frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

- 因為A不是nxn矩陣，只能用高斯喬丹消去法
- 結果：無限多解　$x_1 = t, \quad x_2 = t/3, \quad x_3 = t/3, \quad x_4 = t$
- 最小整數發生在t=3　$x_1 = 3, x_2 = 1, x_3 = 1, x_4 = 3$
- X1=3,　x2=1,　x3=1,　x4=3　$3HCl + Na_3PO_4 \longrightarrow H_3PO_4 + 3NaCl$

6. 用多項式插值法去模擬函數

用多項式插值法去模擬函數



- 若需通過不同 x 軸數值的 n 個點(page. 78)

$$(x_1, y_1), \quad (x_2, y_2), \quad (x_3, y_3), \ldots, \quad (x_n, y_n)$$

- 因為必須符合 n 個條件，所以必須尋找的多項式為

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}$$

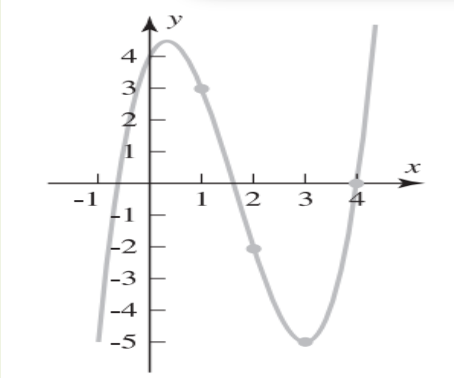
- 因為有n個數據點，所以代入多項式會變成n條方程式

$$\begin{aligned}
a_0 + a_1 x_1 + a_2 x_1^2 + \cdots + a_{n-1} x_1^{n-1} &= y_1 \\
a_0 + a_1 x_2 + a_2 x_2^2 + \cdots + a_{n-1} x_2^{n-1} &= y_2 \\
\vdots \qquad \vdots \qquad \vdots \qquad\qquad\quad \vdots \qquad &\quad \vdots \\
a_0 + a_1 x_n + a_2 x_n^2 + \cdots + a_{n-1} x_n^{n-1} &= y_n
\end{aligned}$$

用多項式插值法去模擬函數

- 因為有n個數據點，所以代入多項式會變成n條方程式(page. 79)

$$
\begin{array}{ccccccc}
a_0 & + a_1 x_1 & + a_2 x_1^2 & + \cdots + & a_{n-1} x_1^{n-1} & = & y_1 \\
a_0 & + a_1 x_2 & + a_2 x_2^2 & + \cdots + & a_{n-1} x_2^{n-1} & = & y_2 \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
a_0 & + a_1 x_n & + a_2 x_n^2 & + \cdots + & a_{n-1} x_n^{n-1} & = & y_n
\end{array}
$$



- 矩陣向量表示式：

$$
\begin{bmatrix}
1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\
1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\
\vdots & \vdots & \vdots & & \vdots \\
1 & x_n & x_n^2 & \cdots & x_n^{n-1}
\end{bmatrix}
\begin{bmatrix} a0 \\ a1 \\ . \\ . \\ an\text{-}1 \end{bmatrix}
=
\begin{bmatrix} y1 \\ y2 \\ . \\ . \\ yn \end{bmatrix}
$$

- 增廣矩陣為

$$
\begin{bmatrix}
1 & x_1 & x_1^2 & \cdots & x_1^{n-1} & y_1 \\
1 & x_2 & x_2^2 & \cdots & x_2^{n-1} & y_2 \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
1 & x_n & x_n^2 & \cdots & x_n^{n-1} & y_n
\end{bmatrix}
$$

範例6：使用高斯–喬登消去法進行多項式插值

1. 求通過下列4點的三次多項式(1, 3), (2, −2), (3, −5), (4, 0)

- 多項式矩陣= $\begin{bmatrix} 1 & x1 & x1^2 & x1^3 \\ 1 & x2 & x2^2 & x2^3 \\ 1 & x3 & x3^2 & x3^3 \\ 1 & x4 & x4^2 & x4^3 \end{bmatrix}\begin{bmatrix} a0 \\ a1 \\ a2 \\ a3 \end{bmatrix} = \begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \end{bmatrix}$

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix}\begin{bmatrix} a0 \\ a1 \\ . \\ . \\ an\text{-}1 \end{bmatrix} = \begin{bmatrix} y1 \\ y2 \\ . \\ . \\ yn \end{bmatrix}$$

- 代入4點：(x1=1, 3), (x2=2, −2), (x3=3, −5), (x4=4, 0)

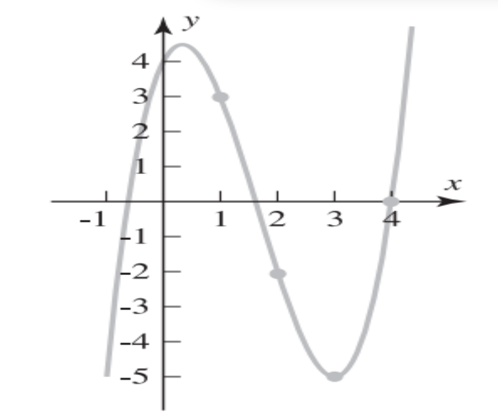
- $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \\ 1 & 4 & 16 & 64 \end{bmatrix}\begin{bmatrix} a0 \\ a1 \\ a2 \\ a3 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ -5 \\ 0 \end{bmatrix}$

(page. 80)

範例6：使用高斯–喬登消去法進行多項式插值

(page.80)

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a0 \\ a1 \\ . \\ . \\ an \end{bmatrix} = \begin{bmatrix} y1 \\ y2 \\ . \\ . \\ yn \end{bmatrix}$$



1. 求通過下列4點的三次多項式(1, 3), (2, −2), (3, −5), (4, 0)

- 擴展矩陣 $= \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 1 & 2 & 4 & 8 & -2 \\ 1 & 3 & 9 & 27 & -5 \\ 1 & 4 & 16 & 64 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 7 & -5 \\ 0 & 2 & 8 & 26 & -8 \\ 0 & 3 & 15 & 63 & -3 \end{bmatrix}$
- $\begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 7 & -5 \\ 0 & 1 & 4 & 13 & -4 \\ 0 & 1 & 5 & 21 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 7 & -5 \\ 0 & 0 & 1 & 6 & 1 \\ 0 & 0 & 2 & 14 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 7 & -5 \\ 0 & 0 & 1 & 6 & 1 \\ 0 & 0 & 0 & 2 & 2 \end{bmatrix}$
- $= \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 3 & 7 & -5 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 0 & 7 & 10 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 4 \\ 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$
- $a0 + a1x1^1 + a2x1^2 + a3x1^3 = \mathbf{4 + 3x1^1 - 5x1^2 + x1^3}$

範例6:Python程式碼

```
from sympy import *
x1,x2,x3,x4 = symbols('x1 x2 x3 x4')
#注意：M矩陣的最右邊=y=轉換後座標
A = Matrix([
    [1,1,1,1,3],
    [1,2,4,8,-2],
    [1,3,9,27,-5],
    [1,4,16,64,0]
    ])
ans = solve_linear_system(A, x1, x2, x3, x4)
print('X=', ans)
```

```
X= {x1: 4, x2: 3, x3: -5, x4: 1}
```

範例 7：用4次多項式近似sin積分

- 近似積分計算(page. 81) $\int_0^1 \sin\left(\frac{\pi x^2}{2}\right) dx$
- 已經通過這些x點 $x_0 = 0, \quad x_1 = 0.25, \quad x_2 = 0.5, \quad x_3 = 0.75, \quad x_4 = 1$
- 已經這些點值 $f(0) = 0, \quad f(0.25) = 0.098017, \quad f(0.5) = 0.382683, \quad f(0.75) = 0.77301, \quad f(1) = 1$
- 用4次多項式近似 $f(x) = \sin\left(\frac{\pi x^2}{2}\right)$
- 擴展矩陣 = $\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a0 \\ a1 \\ . \\ . \\ an \end{bmatrix} = \begin{bmatrix} y1 \\ y2 \\ . \\ . \\ yn \end{bmatrix}$
- 解出係數，得到 $p(x) = 0.098796x + 0.762356x^2 + 2.14429x^3 - 2.00544x^4$

$x_0 = 0, \quad x_1 = 0.25, \quad x_2 = 0.5, \quad x_3 = 0.75, \quad x_4 = 1$

$f(0) = 0, \quad f(0.25) = 0.098017, \quad f(0.5) = 0.382683,$
$f(0.75) = 0.77301, \quad f(1) = 1$

- 多項式矩陣 $= \begin{bmatrix} 1 & \mathrm{x1} & x1^2 & x1^3 \\ 1 & x2 & x2^2 & x2^3 \\ 1 & x3 & x3^2 & x3^3 \\ 1 & x4 & x4^2 & x4^3 \end{bmatrix} \begin{bmatrix} a0 \\ a1 \\ a2 \\ a3 \end{bmatrix}$

- 多項式矩陣 $= \begin{bmatrix} 1 & 0 & 0^2 & 0^3 & 0^4 & 0 \\ 1 & 0.25 & 0.25^2 & 0.25^3 & 0.25^4 & 0.098017 \\ 1 & 0.5 & 0.5^2 & 0.5^3 & 0.5^4 & 0.382683 \\ 1 & 0.75 & 0.75^2 & 0.75^3 & 0.75^4 & 0.77301 \\ 1 & 1 & 1^2 & 1^3 & 1^4 & 1 \end{bmatrix}$

$p(x) = 0.098796x + 0.762356x^2 + 2.14429x^3 - 2.00544x^4$

```
X= {x0: 0, x1: 0.0987960000000001, x2: 0.762356000000001,
x3: 2.14428800000000, x4: -2.00543999999999}
```

Python程式碼

```
X= {x0: 0, x1: 0.0987960000000001, x2: 0.762356000000001,
x3: 2.14428800000000, x4: -2.00543999999999}
```

$$p(x) = 0.098796x + 0.762356x^2 + 2.14429x^3 - 2.00544x^4$$

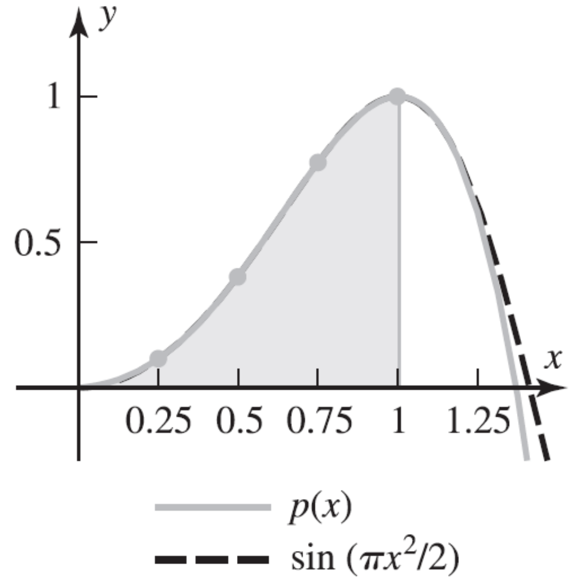
```
from sympy import *
x0,x1,x2,x3,x4 = symbols('x0 x1 x2 x3 x4')
A = Matrix([
    [1,0,0,0,0,0],
    [1,0.25,0.25**2,0.25**3,0.25**4,0.098017],
    [1,0.5,0.5**2,0.5**3,0.5**4,0.382683],
    [1,0.75,0.75**2,0.75**3,0.75**4,0.77301],
    [1,1,1,1,1,1]
    ])
ans = solve_linear_system(A, x0,x1,x2,x3,x4)
print('X=', ans)
```

範例 7：多項式近似積分

- （目標：要計算積分） $\int_0^1 \sin\left(\frac{\pi x^2}{2}\right) dx$
- 方法：針對多項式積分(0,1)

$$p(x) = 0.098796x + 0.762356x^2 + 2.14429x^3 - 2.00544x^4$$

- 結果： $\int_0^1 p(x)\,dx \approx 0.438501$



➲ 圖 1.9.13

- 多項式近似函數，數值差不多
- (page. 81)

重要

行列式的重點摘要-1

- 1. 行列式非常實用，且重要
- 2. 行列式功用1：預先判斷聯立方程式是否有唯一解
  - 若det(A)≠0，則系統有唯一解，即可用高斯消去法求解
- 3. 行列式功用2：預先判斷反矩陣是否存在
  - 若det(A)≠0，則反矩陣存在，即可用公式計算反矩陣
- 4. 矩陣A的行列式值det(A)所代表的物理意義
  - 2D矩陣：代表座標轉換後，面積的縮放率
  - 3D矩陣：代表座標轉換後，體積的縮放率

重要

行列式的重點摘要-2

- 5. 2D行列式特殊公式：

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

- 6. 3D行列式特殊公式：

$$\begin{vmatrix} 1 & 2 & 3 \\ -4 & 5 & 6 \\ 7 & -8 & 9 \end{vmatrix} = \left|\begin{array}{ccc|cc} 1 & 2 & 3 & 1 & 2 \\ -4 & 5 & 6 & -4 & 5 \\ 7 & -8 & 9 & 7 & -8 \end{array}\right.$$

$$= [45 + 84 + 96] - [105 - 48 - 72]$$

行列式的重點摘要-3

- 7. 4D以上行列式公式：餘因子法
  - 找任何一列（有0的最好，因為可以省略計算）
  - $= \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} - \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} + \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix}$
  - =

方法二：任何一行或一列展开

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} \xrightarrow{\text{按第一行展开}}$$

$$= a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

$$= a(ei - fh) - b(di - fg) + c(dh - eg)$$

重要

判別是否有唯一解的方法有兩種

- (1). 方法1：用增廣矩陣，將之化簡：簡化列梯形，然後評估每個變數是否存在？
  - 缺點：速度慢，高斯消去法，計算到最後才知道
  - 優點：可以判別是無解，還是無限多組解
- (2). 方法2：用原始矩陣A的行列式det(A)判別
  - 解聯立方程式之前要先判斷是否有唯一解=> det(A)≠
  - 若det(A)≠0，則系統有唯一解
  - 若det(A)=0，則系統可能無解，可能無限多組解
    - 若det(A)=0，且det(A|B)=0，則無限多組解
    - 若det(A)=0，且det(A|B)≠0，則無解

(a) $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ \boxed{0} & \boxed{0} & \boxed{0} & \boxed{1} \end{bmatrix}$ 無解

(b) $\begin{bmatrix} 1 & 0 & 3 & -1 \\ 0 & 1 & -4 & 2 \\ \boxed{0} & \boxed{0} & \boxed{0} & \boxed{0} \end{bmatrix}$ 無限多組解

解聯立方程式的重點摘要-1

- 1. 解聯立方程式之前要先判斷是否有唯一解 => det(A)≠0
  - 因為克拉瑪法則解x1, x2..的公式為

$$x_1 = \frac{\det(A_1)}{\det(A)} = \frac{-40}{44} = \frac{-10}{11}, \qquad x_2 = \frac{\det(A_2)}{\det(A)} = \frac{72}{44} = \frac{18}{11},$$

  - 若是det(A)=0，則系統無解，或是無限多組解
- 2. 求解聯立方程式，求解反矩陣，都是要先判斷det(A)≠0，才有唯一解

解聯立方程式的重點摘要-2

重要

- 3.計算聯立方程式的方法有三種：
  - (1).高斯消去法
  - (2).Cramer’s rule克拉瑪法則

    A必須為nxn矩陣
  - (3).反矩陣法

    A必須為nxn矩陣

重要

解聯立方程式的重點摘要-3

- 4. 計算聯立方程式的方法有三種：
  - (1). 高斯消去法（用擴展矩陣）
  - (2). 反矩陣法
    - $A^{-1}=\frac{伴隨矩陣}{det(A)}=\frac{餘因子矩陣^{T}}{det(A)}$ （餘因子矩陣要注意正負號）
    - 解聯立方程式：$\vec{x}=A^{-1}\vec{v}$
  - (3). Cramer’s rule克拉瑪法則

用高斯喬丹消去法，結果

$$A^{-1}=\left[\begin{array}{ccc|ccc}1&0&0&-40&16&9\\0&1&0&13&-5&-3\\0&0&1&5&-2&-1\end{array}\right]$$

$$x_1=\frac{\det(A_1)}{\det(A)},\quad x_2=\frac{\det(A_2)}{\det(A)},\ldots,\quad x_n=\frac{\det(A_n)}{\det(A)}$$