第二章、研究方法與理論

本章將介紹本篇論文所用到的重要方法與理論。首先在 2-1 節中，我們將根據機器人學理論，推導出本論文的舉重機器手臂所用到的靜力與動力方程式。2-2 節介紹本篇論文最重要的理論—Dijkstra 演算法，說明並討論它的基本原理與重要的性質。在 2-3 節中將介紹 B-spline Curve 與 Curve Fitting 的一些基本理論。最後在 2-4 節，我們將介紹 3D 模擬程式常用到的 OpenGL 函式庫的簡介。

2.1 機器人學

由於當動力方程式中的所有角速度與角加速度都為零時，動力方程式即變為靜力方程式。所以在本節中我們將直接推導動力方程式，最後再將具角速度與角加速度的項去掉，即可另外求得靜力方程式。

2.1.1 機器人動力學

我們首先必須建立機器人的連桿座標系，並定義相關參數。圖 2.1 為 Denavit-Hartenberg 方法描述各連桿間轉換矩陣中所用到的參數定義，其參數說明綜合如下：

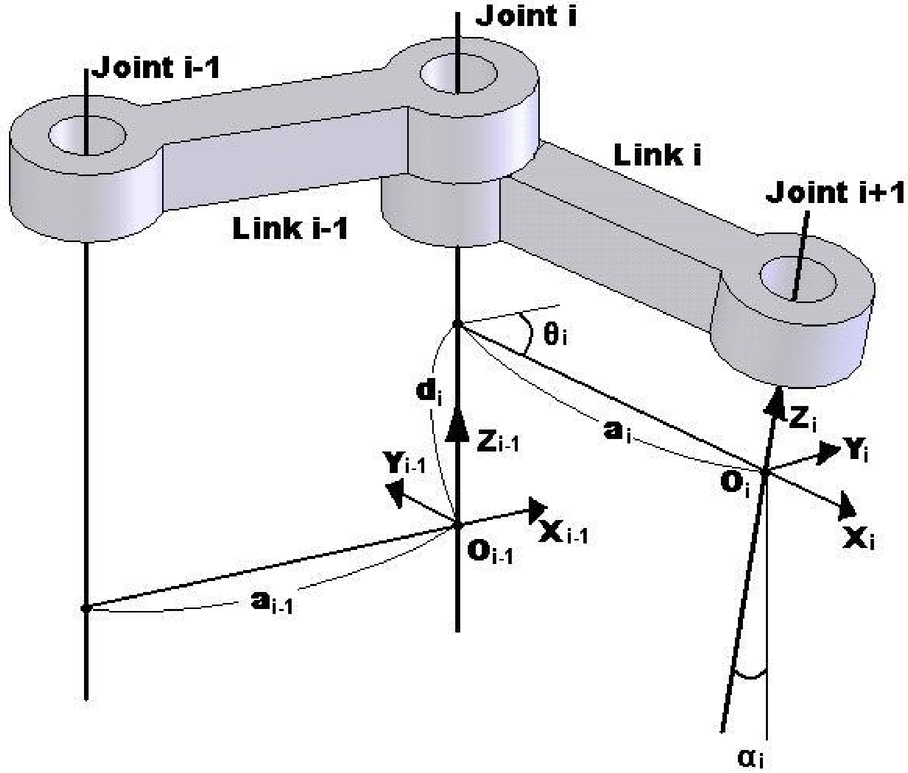

圖 2.1 Denavit-Hartenberg 參數定義圖

1. O_i：第 i 座標系的原點位置。

2. Z_i：為第 i 座標系的軸線向量。

3. X_i：同時正交 Z_i 與 Z_{i-1} 的軸線向量，方向為由 Z_{i-1} 指向 Z_i。

4. Y_i：同時正交 Z_i 與 X_i 的軸線向量。

5. a_{i-1}：沿 X_{i-1} 軸所得的 Z_i 軸、Z_{i-1} 軸間之距離。

6. α_{i-1}：繞 X_i 軸所得的 Z_i 與 Z_{i-1} 間之夾角。

6. d_i：沿 Z_{i-1} 軸所得的 X_i 軸與 X_{i-1} 軸間之距離。

7. θ_i：繞 Z_{i-1} 軸所得的 X_i 軸與 X_{i-1} 軸間之夾角，即第 i 座標系相對於第 i-1 座標系的旋轉角度。

8. ${}^{ci}I_i$：相對於第 i 桿質量中心的慣性張量。

9. ${}^{i-1}r_i$：第 i 座標系相對於第 i-1 座標系的位置向量，即$\overrightarrow{O_{i-1}O_i}$。

10. ${}^{i}r_{ci}$：第 i 連桿的質量中心相對於第 i 座標系的位置向量。

11. $\dot{\theta}_i$：第 i 座標系的角速度。

12. $\ddot{\theta}_i$：第 i 座標系的角加速度。

13. ${}^{i}\omega_i$：連桿 i 相對於第 i 座標系的角速度。

14. ${}^{i}\dot{\omega}_i$：連桿 i 相對於第 i 座標系的角加速度。

15. ${}^{i}\dot{v}_i$：第 i 座標系的線加速度。

16. ${}^{i}\dot{v}_{ci}$：第 i 桿的質量中心相對於第 i 座標系的線加速度。

17. ${}^{i}F_i$：作用於第 i 桿質量中心的淨力。

18. ${}^{i}f_i$：作用於第 i 座標系的淨力。

19. ${}^{i}N_i$：作用於第 i 桿質量中心的淨力矩。

20. ${}^{i}n_i$：作用於第 i 座標系的淨力矩。

21. ${}^{i}R_{i+1}$：第 i+1 座標系相對於第 i 座標系的旋轉矩陣。

(1)順向運算

我們首先計算每一連桿的角速度、角加速度、線速度和線加速度。這些物理量可以從總系統的原點推導到連桿的手端。對於系統的原點，我們通常令其速度為零，即${}^{0}\omega_0={}^{0}v_0={}^{0}\dot{\omega}_0={}^{0}\dot{v}_0=0$，則每一軸與連桿的速度與加速度計算公式如下：

$$ {}^{i}\omega_i = {}^{i}R_i\left({}^{i-1}\omega_{i-1} + {}^{i-1}z_{i-1}\,\dot{\theta}_i \right) \tag{2-1} $$

$$ {}^{i}\dot{\omega}_i = {}^{i}R_{i-1}\left({}^{i-1}\dot{\omega}_{i-1} + {}^{i-1}z_{i-1}\,\ddot{\theta}_i + {}^{i-1}\omega_{i-1} \times {}^{i-1}z_{i-1}\,\dot{\theta}_i \right) \tag{2-2} $$

$$ {}^{i}\dot{v}_i = {}^{i}R_{i-1}\left[{}^{i}\dot{\omega}_i \times {}^{i-1}r_i + {}^{i-1}\omega_{i-1} \times ({}^{i-1}\omega_{i-1} \times {}^{i-1}r_i) + {}^{i-1}\dot{v}_{i-1} \right] \tag{2-3} $$

$$ {}^{i}\dot{v}_{ci} = {}^{i}\dot{\omega}_i \times {}^{i}r_{ci} + {}^{i}\omega_i \times ({}^{i}\omega_i \times {}^{i}r_{ci}) + {}^{i}\dot{v}_i \tag{2-4} $$

其中：

$$ {}^{i-1}r_i = \begin{bmatrix} a_i \\ 0 \\ 0 \end{bmatrix} \quad \text{且} \quad {}^{i}r_{ci} = \begin{bmatrix} -a_i \\ 0 \\ 0 \end{bmatrix} \tag{2-5} $$

$$ {}^{i}R_{i-1} = \begin{bmatrix} c\theta_i & s\theta_i & 0 \\ -c\alpha_i s\theta_i & c\alpha_i c\theta_i & s\alpha_i \\ s\alpha_i s\theta_i & -s\alpha_i c\theta_i & c\alpha_i \end{bmatrix} \tag{2-6} $$

(2) 反向運算

一旦每一連桿與軸的速度與加速度決定之後，我們可以開始計算每一連桿與軸的力與力矩大小。這些物理量的推導為反過來從連桿的手端到總系統的原點。

$$ {}^{i}F_i = m_i\,{}^{i}\dot{v}_{ci} \tag{2-7} $$

$$ {}^{i}N_i = {}^{ci}I_i\,{}^{i}\dot{\omega}_i + {}^{i}\omega_i \times {}^{ci}I_i\,{}^{i}\omega_i \tag{2-8} $$

$$ {}^{i}f_i = {}^{i}R_{i+1}\,{}^{i+1}f_{i+1} + {}^{i}F_i \tag{2-9} $$

$$ {}^{i}n_i = {}^{i}N_i + {}^{i}R_{i+1}\,{}^{i+1}n_{i+1} + {}^{i}r_{ci} \times {}^{i}F_i + {}^{i}r_{i+1} \times {}^{i}R_{i+1}\,{}^{i+1}f_{i+1} \tag{2-10} $$

其中：

$$
{}^{i}R_{i+1}=\begin{bmatrix} c\theta_{i+1} & s\theta_{i+1} & 0 \\ -c\alpha_{i+1}s\theta_{i+1} & c\alpha_{i+1}c\theta_{i+1} & s\alpha_{i+1} \\ s\alpha_{i+1}s\theta_{i+1} & -s\alpha_{i+1}c\theta_{i+1} & c\alpha_{i+1} \end{bmatrix} \tag{2-11}
$$

2.1.2 動力方程式的推導

接著我們開始推導平面運動之三連桿的動力方程式。這裡我們假設每一連桿的長度為 L_i ，密度均勻且總質量為 m_i ，(這裡各連桿的質量中心假設位於 1/2 連桿長度處)。其它參數如載重 M 與各軸角度 θ_1 、θ_2 、θ_3 的表示如圖 2.1。對於三軸機器手臂的 Denavit-Hartenberg 參數如下表 2-1：

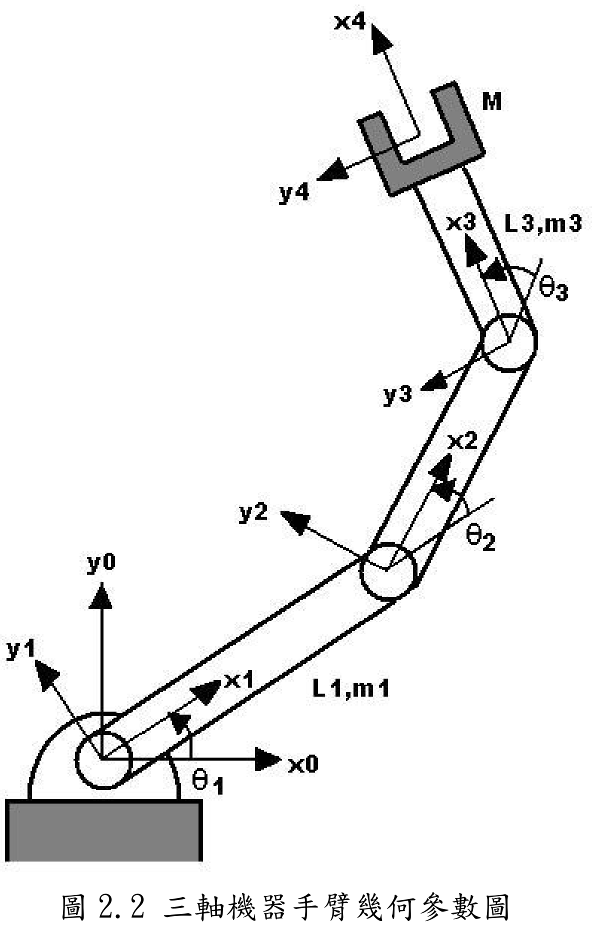

圖 2.2 三軸機器手臂幾何參數圖

表 2.1 三軸機器手臂的 Denavit-Hartenberg 參數表

Joint	α_i	a_i	d_i	θ_i
1	0	L1	0	θ_1
2	0	L2	0	θ_2
3	0	L3	0	θ_3

各連桿相對於質心之慣性張量分別如下：

$$^{c1}I_1 = \frac{m_1 l_1^{\ 2}}{12}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-12}$$

$$^{c2}I_2 = \frac{m_2 l_2^{\ 2}}{12}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-13}$$

$$^{c3}I_3 = \frac{m_3 l_3^{\ 2}}{12}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-14}$$

而每一連桿之頂端與質心的位置向量 $^i r_i$ 和 $^i r_{ci}$ 分別如下：

$$^1 r_1 = \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \quad 且 \quad ^1 r_{c1} = \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \tag{2-15}$$

$$^2 r_2 = \begin{bmatrix} l_2 \\ 0 \\ 0 \end{bmatrix} \quad 且 \quad ^2 r_{c2} = \begin{bmatrix} l_2 \\ 0 \\ 0 \end{bmatrix} \tag{2-16}$$

$$^3 r_3 = \begin{bmatrix} l_3 \\ 0 \\ 0 \end{bmatrix} \quad 且 \quad ^3 r_{c3} = \begin{bmatrix} l_3 \\ 0 \\ 0 \end{bmatrix} \tag{2-17}$$

(a)順向運算：接著我們首先計算第一連桿的速度與加速度。由於 ${}^0\omega_0={}^0v_0={}^0\dot{\omega}_0={}^0\dot{v}_0=0$，所以我們可以得到：

$$ {}^1\omega_1=\begin{bmatrix}0\\0\\\dot{\theta}_1\end{bmatrix} \quad ; \quad {}^1\dot{\omega}_1=\begin{bmatrix}0\\0\\\ddot{\theta}_1\end{bmatrix} \tag{2-18}$$

$$ {}^1\dot{v}_1={}^1R_0\left[{}^0\dot{\omega}_0\times{}^0r_1+{}^0\omega_0\times({}^0\omega_0\times{}^0r_1)+{}^0\dot{v}_0\right]$$

$$=\begin{bmatrix}c_1 & s_1 & 0\\-s_1 & c_1 & 0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}0\\g\\0\end{bmatrix}=\begin{bmatrix}gs_1\\gc_1\\0\end{bmatrix} \tag{2-19}$$

$$ {}^1\dot{v}_{c1}={}^1\dot{\omega}_1\times{}^1r_{c1}+{}^1\omega_1\times({}^1\omega_1\times{}^1r_{c1})+{}^1\dot{v}_1$$

$$=\begin{bmatrix}0\\\frac{l_1}{2}\ddot{\theta}_1\\0\end{bmatrix}+\begin{bmatrix}-\frac{l_1}{2}\dot{\theta}_1^2\\0\\0\end{bmatrix}+\begin{bmatrix}gs_1\\gc_1\\0\end{bmatrix}=\begin{bmatrix}-\frac{l_1}{2}\dot{\theta}_1^2+gs_1\end{bmatrix} \tag{2-20}$$

$$ {}^1F_1=m_1\,{}^1\dot{v}_{c1}=\begin{bmatrix}-\frac{m_1}{2}l_1\dot{\theta}_1^2+m_1gs_1\\\frac{m_1}{2}l_1\ddot{\theta}_1+m_1gc_1\\0\end{bmatrix} \tag{2-21}$$

$$ {}^1N_1={}^{c1}I_1\,{}^1\dot{\omega}_1+{}^1\omega_1\times{}^{c1}I_1\,{}^1\omega_1=\begin{bmatrix}0\\0\\\frac{m_1l_1^2}{12}\ddot{\theta}_1\end{bmatrix} \tag{2-22}$$

接著我們計算第二連桿的速度與加速度，我們可以得到：

$$
{}^{2}\omega_{2}=\begin{bmatrix}0\\0\\\dot{\theta}_{1}+\dot{\theta}_{2}\end{bmatrix}\quad;\quad {}^{2}\dot{\omega}_{2}=\begin{bmatrix}0\\0\\\ddot{\theta}_{1}+\ddot{\theta}_{2}\end{bmatrix} \tag{2-23}
$$

$$
{}^{2}\dot{v}_{2}={}^{2}R_{1}\left[{}^{1}\dot{\omega}_{1}\times{}^{1}r_{2}+{}^{1}\omega_{1}\times({}^{1}\omega_{1}\times{}^{1}r_{2})+{}^{1}\dot{v}_{1}\right]
$$

$$
=\begin{bmatrix}l_{1}\ddot{\theta}_{1}s_{2}-l_{1}\dot{\theta}_{1}^{2}c_{2}+gs_{12}\\l_{1}\ddot{\theta}_{1}c_{2}+l_{1}\dot{\theta}_{1}^{2}s_{2}+gc_{12}\\0\end{bmatrix} \tag{2-24}
$$

$$
{}^{2}\dot{v}_{c2}={}^{2}\dot{\omega}_{2}\times{}^{2}r_{c2}+{}^{2}\omega_{2}\times({}^{2}\omega_{2}\times{}^{2}r_{c2})+{}^{2}\dot{v}_{2}
$$

$$
=\begin{bmatrix}-\frac{l_{1}}{2}\left(\dot{\theta}_{1}+\dot{\theta}_{2}\right)^{2}+l_{1}\ddot{\theta}_{1}s_{2}-l_{1}\dot{\theta}_{1}^{2}c_{2}+gs_{12}\\\frac{l_{1}}{2}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}\right)+l_{1}\ddot{\theta}_{1}c_{2}+l_{1}\dot{\theta}_{1}^{2}s_{2}+gc_{12}\\0\end{bmatrix} \tag{2-25}
$$

$$
{}^{2}F_{2}=m_{2}\,{}^{2}\dot{v}_{c2}
$$

$$
=\begin{bmatrix}-\frac{m_{2}}{2}l_{2}\left(\dot{\theta}_{1}+\dot{\theta}_{2}\right)^{2}+m_{2}l_{1}\ddot{\theta}_{1}s_{2}-m_{2}l_{1}\dot{\theta}_{1}^{2}c_{2}+m_{2}gs_{12}\\\frac{m_{2}}{2}l_{2}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}\right)+m_{2}l_{1}\ddot{\theta}_{1}c_{2}+m_{2}l_{1}\dot{\theta}_{1}^{2}s_{2}+m_{2}gc_{12}\\0\end{bmatrix} \tag{2-26}
$$

$$ {}^2N_2 = {}^{c2}I_2\,{}^2\dot{\omega}_2 + {}^2\omega_2 \times {}^{c2}I_2\,{}^2\omega_2 = \begin{bmatrix} 0 \\ 0 \\ \frac{m_2 l_2^2}{12}\left(\ddot{\theta}_1 + \ddot{\theta}_2\right) \end{bmatrix} \tag{2-27} $$

最後計算第三連桿的速度與加速度，我們可以得到：

$$ {}^3\omega_3 = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 \end{bmatrix} \quad ; \quad {}^3\dot{\omega}_3 = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3 \end{bmatrix} \tag{2-28} $$

$$ {}^3\dot{v}_3 = {}^3R_2\left[{}^2\dot{\omega}_2 \times {}^2r_3 + {}^2\omega_2 \times ({}^2\omega_2 \times {}^2r_3) + {}^2\dot{v}_2 \right] $$

$$ = \begin{bmatrix} l_1\ddot{\theta}_1 s_{23} - l_1\dot{\theta}_1^2 c_{23} - l_2 c_3\left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + l_2 s_3\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) \\ + g s_{123} \\ l_1\ddot{\theta}_1 c_{23} + l_1\dot{\theta}_1^2 s_{23} + l_2 s_3\left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + l_2 c_3\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) \\ + g c_{123} \\ 0 \end{bmatrix} \tag{2-29} $$

$${}^{3}\dot{v}_{c3} = {}^{3}\dot{\omega}_{3} \times {}^{3}r_{c3} + {}^{3}\omega_{3} \times ({}^{3}\omega_{3} \times {}^{3}r_{c3}) + {}^{3}\dot{v}_{3}$$

$$= \begin{bmatrix} l_1 \ddot{\theta}_1 s_{23} - l_1 \dot{\theta}_1^{2} c_{23} - l_2 c_3 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + l_2 s_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) \\ -\frac{l_3}{2}\left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2 + g s_{123} \\ \\ l_1 \ddot{\theta}_1 c_{23} + l_1 \dot{\theta}_1^{2} s_{23} + l_2 s_3 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + l_2 c_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) \\ +\frac{l_3}{2}\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) + g c_{123} \\ \\ 0 \end{bmatrix} \qquad (2\text{-}30)$$

$${}^{3}F_{3} = m_3 \, {}^{3}\dot{v}_{c3} + M \, {}^{3}\dot{v}_{3}$$

$$= \begin{bmatrix} (m_3 + M) l_1 \ddot{\theta}_1 s_{23} - (m_3 + M) l_1 \dot{\theta}_1^{2} c_{23} - (m_3 + M) l_2 c_3 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 \\ +(m_3 + M) l_2 s_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) - (\frac{m3}{2} + M) l_3 \left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2 \\ +(m_3 + M) g s_{123} \\ \\ (m_3 + M) l_1 \ddot{\theta}_1 c_{23} + (m_3 + M) l_1 \dot{\theta}_1^{2} s_{23} + (m_3 + M) l_2 s_3 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 \\ +(m_3 + M) l_2 c_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) + (\frac{m3}{2} + M) l_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) \\ +(m_3 + M) g c_{123} \\ \\ 0 \end{bmatrix} \qquad (2\text{-}31)$$

$$
{}^{3}N_{3}={}^{c3}I_{3}\,{}^{3}\dot{\omega}_{3}+{}^{3}\omega_{3}\times{}^{c3}I_{3}\,{}^{3}\omega_{3}=\begin{bmatrix}0\\0\\\dfrac{m_{3}l_{3}^{2}}{12}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}+\ddot{\theta}_{3}\right)\end{bmatrix} \tag{2-32}
$$

(b)反向運算：對於力與力矩的計算，我們首先計算第三連桿的力與力矩值；接著計算第二連桿的力與力矩值，最後是第一連桿的力與力矩值；因此首先我們計算第三連桿的力與力矩值。由於${}^{4}f_{4}=0$，所以我們可以得到：

$$
{}^{3}f_{3}={}^{3}R_{4}\,{}^{4}f_{4}+{}^{3}F_{3}={}^{3}F_{3} \tag{2-33}
$$

$$
\begin{aligned}
{}^{3}n_{3}&={}^{3}N_{3}+{}^{3}R_{4}\,{}^{4}n_{4}+{}^{3}r_{c3}\times{}^{3}F_{3}+{}^{3}r_{4}\times{}^{3}R_{4}\,{}^{4}f_{4}\\
&=\begin{bmatrix}0\\0\\
\dfrac{m_{3}l_{3}^{2}}{12}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}+\ddot{\theta}_{3}\right)+\left(\dfrac{m_{3}}{2}+M\right)l_{1}l_{3}\ddot{\theta}_{1}c_{23}+\left(\dfrac{m_{3}}{2}+M\right)l_{3}gc_{123}\\
+\left(\dfrac{m_{3}}{2}+M\right)l_{1}l_{3}\dot{\theta}_{1}^{2}s_{23}+\left(\dfrac{m_{3}}{2}+M\right)l_{2}l_{3}s_{3}\left(\dot{\theta}_{1}+\dot{\theta}_{2}\right)^{2}\\
+\left(\dfrac{m_{3}}{2}+M\right)l_{2}l_{3}c_{3}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}\right)+\left(\dfrac{m_{3}}{4}+M\right)l_{3}^{2}\left(\ddot{\theta}_{1}+\ddot{\theta}_{2}+\ddot{\theta}_{3}\right)
\end{bmatrix}
\end{aligned} \tag{2-34}
$$

接著我們計算第二連桿的力與力矩值，我們可以得到：

$$
{}^{2}f_{2}={}^{2}R_{3}\,{}^{3}f_{3}+{}^{2}F_{2}
$$

$$
=\begin{bmatrix}
\begin{aligned}
&-\frac{m_2}{2}l_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2+m_2l_1\ddot{\theta}_1 s_2-m_2l_1\dot{\theta}_1^2 c_2+(\frac{m_2}{2})l_2gs_{12}\\
&+\left(m_3+M\right)l_1\ddot{\theta}_1 s_2-\left(m_3+M\right)l_1\dot{\theta}_1^2 c_2-\left(m_3+M\right)l_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2\\
&-\left(\frac{m_3}{2}+M\right)l_3c_3\left(\dot{\theta}_1+\dot{\theta}_2+\dot{\theta}_3\right)^2-\left(\frac{m_3}{2}+M\right)l_3s_3\left(\ddot{\theta}_1+\ddot{\theta}_2+\ddot{\theta}_3\right)\\
&+\left(m_3+M\right)l_2gs_{12}
\end{aligned}\\[2ex]
\begin{aligned}
&\frac{m_2}{2}l_2\left(\ddot{\theta}_1+\ddot{\theta}_2\right)+m_2l_1\ddot{\theta}_1 c_2+m_2l_1\dot{\theta}_1^2 s_2+(\frac{m_2}{2})l_2gc_{12}\\
&+\left(m_3+M\right)l_1\ddot{\theta}_1 c_2+\left(m_3+M\right)l_1\dot{\theta}_1^2 s_2+\left(m_3+M\right)l_2\left(\ddot{\theta}_1+\ddot{\theta}_2\right)\\
&-\left(\frac{m_3}{2}+M\right)l_3s_3\left(\dot{\theta}_1+\dot{\theta}_2+\dot{\theta}_3\right)^2+\left(\frac{m_3}{2}+M\right)l_3c_3\left(\ddot{\theta}_1+\ddot{\theta}_2+\ddot{\theta}_3\right)\\
&+\left(m_3+M\right)l_2gc_{12}
\end{aligned}\\[2ex]
0
\end{bmatrix}
\qquad (2\text{-}35)
$$

$$
{}^{2}n_{2}={}^{2}N_{2}+{}^{2}R_{3}\,{}^{3}n_{3}+{}^{2}r_{c2}\times{}^{2}F_{2}+{}^{2}r_{3}\times{}^{2}R_{3}\,{}^{3}f_{3}
$$

$$
=\begin{bmatrix}
0\\[2ex]
0\\[2ex]
\begin{aligned}
&\frac{m_2l_2^2}{12}\left(\ddot{\theta}_1+\ddot{\theta}_2\right)+\frac{m_3l_3^2}{12}\left(\ddot{\theta}_1+\ddot{\theta}_2+\ddot{\theta}_3\right)+\left(\frac{m_3}{2}+M\right)l_1l_3\ddot{\theta}_1 c_{23}\\
&+\left(\frac{m_3}{2}+M\right)l_3gc_{123}+\left(\frac{m_3}{2}+M\right)l_1l_3\dot{\theta}_1^2 s_{23}\\
&+\left(\frac{m_3}{2}+M\right)l_2l_3s_3\left(\dot{\theta}_1+\dot{\theta}_2\right)^2+\left(\frac{m_3}{2}+M\right)l_2l_3c_3\left(\ddot{\theta}_1+\ddot{\theta}_2\right)\\
&+\left(\frac{m_3}{4}+M\right)l_3^2\left(\ddot{\theta}_1+\ddot{\theta}_2+\ddot{\theta}_3\right)+\frac{m_2l_2^2}{4}\left(\ddot{\theta}_1+\ddot{\theta}_2\right)+\frac{m_2}{2}l_1l_2\ddot{\theta}_1 c_2\\
&+\frac{m_2}{2}l_1l_2\dot{\theta}_1^2 s_2+\frac{m_2}{2}l_2gc_{12}+\left(m_3+M\right)l_1l_2\ddot{\theta}_1 c_2+\left(m_3+M\right)l_2gc_{12}\\
&+\left(m_3+M\right)l_1l_2\dot{\theta}_1^2 s_2+\left(m_3+M\right)l_2^2\left(\ddot{\theta}_1+\ddot{\theta}_2\right)\\
&-\left(\frac{m_3}{2}+M\right)l_2l_3s_3\left(\dot{\theta}_1+\dot{\theta}_2+\dot{\theta}_3\right)^2+\left(\frac{m_3}{2}+M\right)l_2l_3c_3\left(\ddot{\theta}_1+\ddot{\theta}_2+\ddot{\theta}_3\right)
\end{aligned}
\end{bmatrix}
\qquad (2\text{-}36)
$$

最後我們計算第一連桿的力與力矩值，我們可以得到：

$$
{}^{1}n_{1} = {}^{1}N_{1} + {}^{1}R_{2}\,{}^{2}n_{2} + {}^{1}r_{c1} \times {}^{1}F_{1} + {}^{1}r_{2} \times {}^{1}R_{2}\,{}^{2}f_{2}
$$

$$
= \begin{bmatrix}
0 \\
0 \\
\frac{m_1 l_1^2}{12}\ddot{\theta}_1 + \frac{m_2 l_2^2}{12}\left(\ddot{\theta}_1 + \ddot{\theta}_2\right) + \frac{m_3 l_3^2}{12}\left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) + (\frac{m_3}{2} + M)l_1 l_3 \ddot{\theta}_1 c_{23} \\
+(\frac{m_3}{2} + M)l_3 g c_{123} + (\frac{m_3}{2} + M)l_1 l_3 \dot{\theta}_1^2 s_{23} + (\frac{m_3}{2} + M)l_2 l_3 s_3 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 \\
+(\frac{m_3}{2} + M)l_2 l_3 c_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) + (\frac{m_3}{4} + M)l_3^2 \left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) \\
+ \frac{m_2 l_2^2}{4}\left(\ddot{\theta}_1 + \ddot{\theta}_2\right) + \frac{m_2}{2} l_1 l_2 \ddot{\theta}_1 c_2 + \frac{m_2}{2} l_1 l_2 \dot{\theta}_1^2 s_2 + \frac{m_2}{2} l_2 g c_{12} \\
+(m_3 + M)l_1 l_2 \ddot{\theta}_1 c_2 + (m_3 + M)l_2 g c_{12} + (m_3 + M)l_1 l_2 \dot{\theta}_1^2 s_2 \\
+(m_3 + M)l_2^2 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) - (\frac{m_3}{2} + M)l_2 l_3 s_3 \left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2 \\
+(\frac{m_3}{2} + M)l_2 l_3 c_3 \left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right) + \frac{m_1}{4} l_1^2 \ddot{\theta}_1 + \frac{m_1}{2} l_1 g c_1 \\
+ \frac{m_2}{2} l_1 l_2 c_2 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) - \frac{m_2}{2} l_1 l_2 s_2 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 + m_2 l_1^2 \ddot{\theta}_1 + m_2 l_1 g c_1 \\
+(m_3 + M)l_1^2 \ddot{\theta}_1 + (m_3 + M)l_1 g c_1 + (m_3 + M)l_1 l_2 c_2 \left(\ddot{\theta}_1 + \ddot{\theta}_2\right) \\
-(m_3 + M)l_1 l_2 s_2 \left(\dot{\theta}_1 + \dot{\theta}_2\right)^2 - (\frac{m_3}{2} + M)l_1 l_3 c_{23} \left(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3\right)^2 \\
+(\frac{m_3}{2} + M)l_1 l_3 c_{23} \left(\ddot{\theta}_1 + \ddot{\theta}_2 + \ddot{\theta}_3\right)
\end{bmatrix} \qquad (2\text{-}37)
$$

2.2 Dijkstra 演算法

Dijkstra演算法是由Edsger Wybe Dijkstra[6][7][8]於1959年提出的一個最短路徑演算法，也是目前公認求解最短路徑的高效經典演算法之一。例如Cheng等人[9]曾採用Dijkstra演算法，求解規劃任意三次元曲面兩點間最短路徑的問題。Dijkstra演算法可以保證求出某一節點到其他所有節點的最短路徑。目前最常用的最短路徑演算法－A*演算法，其實也是

由Dijkstra演算法衍生而來。Dijkstra演算法與A*演算法的優缺點在於：Dijkstra演算法具有最佳答案的保證性，但搜尋的空間與時間複雜度卻很大；反之，A*演算法的空間與時間複雜度相對於Dijkstra演算法要小很多，但卻不能保證搜尋出的答案為最佳的。為了綜合這兩種方法的優缺點，我們將在第四章提出修正Dijkstra演算法的新方法。

Dijkstra 演算法的主要特點是以起始點為中心向外一層一層擴展，直到擴展到終點為止，所以 Dijkstra 演算法能夠求得最短路徑的最佳答案。Dijkstra 演算法一般對於節點的紀錄有兩種方式，一種用永久和臨時標號方式，一種是用 OPEN、CLOSE 表方式，在本篇論文中均採用 OPEN、CLOSE 表的方式。其搜尋過程簡介如下所述：

1. 建立兩個節點紀錄－OPEN 與 CLOSE。OPEN 表紀錄所有已生成而未選取過的節點，CLOSED 表中記錄已選取過的節點。
2. 計算起點的目標函數值，並把它放入 OPEN 表中等待檢查。
3. 從 OPEN 表中找出目標函數最小的節點，找出這個節點的所有子節點，並計算這些子節點的目標函數值。
   (a)若子節點已經在 OPEN 表中了，比較子節點新舊目標函數值，保留目標函數值較小的子節點資料。
   (b)若子節點已經在 CLOSE 表中了，比較子節點新舊目標函數值。若子節點的新目標函數值較小，更新子節點資料並重新加入 OPEN 表

中。

(c)不在 OPEN 表也不在 CLOSE 表中，則直接將子節點加入 OPEN 表中。

4. 重複第 3 步。直到挑選的節點為終點則完成搜尋；或是 OPEN 表裡沒有節點則表示沒有路徑可以到終點。

上述的節點資料包括：(1)節點的座標(2)節點的父節點座標：因為等到搜尋結束時，可以從終點回溯父節點到起點以得到路徑。(3)節點的目標函數值：每個節點的目標函數值為從父節點累加，所以當父節點改變時，節點可能會有不一樣的目標函數值。下面我們將舉一個簡單的例子示範 Dijkstra 演算法。如下圖 2.3 為一個3×3的 2D 網格圖，每個網格到鄰近網格的距離均為 1。欲求 S 點到 G 點的最短路徑，以 Dijkstra 演算法求解的步驟如下。在下列參考範例中的(A_a, B)，A 表示節點編號，a表示 A 的父節點編號，B 表示節點 A 目前的目標函數值：

S	1	4
2	3	6
5	7	G

圖 2.3　3×3的 2D 網格圖

(1) 挑選：(S,0)

子節點：$(1_S, 1), (2_S, 1), (3_S, 1)$

OPEN 表：$(1_S, 1), (2_S, 1), (3_S, 1)$

CLOSE 表：(S,0)

(2) 挑選：$(1_S, 1)$

子節點：$(S_1, 1), (2_1, 2), (3_1, 2), (4_1, 2), (6_1, 2)$

OPEN 表：$(2_S, 1), (3_S, 1), (4_1, 2), (6_1, 2)$

CLOSE 表：(S,0), $(1_S, 1)$

(3) 挑選：$(2_S, 1)$

子節點：$(S_2, 2), (1_2, 2), (3_2, 2), (5_2, 2), (7_2, 2)$

OPEN 表：$(3_S, 1), (4_1, 2), (5_2, 2), (6_1, 2), (7_2, 2)$

CLOSE 表：(S,0), $(1_S, 1), (2_S, 1)$

(4) 挑選：$(3_S, 1)$

子節點：$(S_3, 2), (1_3, 2), (2_3, 2), (4_3, 2), (5_3, 2),$

$(6_3, 2), (7_3, 2), (G_3, 2)$

OPEN 表：$(4_1, 2), (5_2, 2), (6_1, 2), (7_2, 2), (G_3, 2)$

CLOSE 表：(S,0), $(1_S, 1), (2_S, 1), (3_S, 1)$

(5) 挑選：$(4_1, 2)$

子節點：$(1_4, 3), (3_4, 3), (6_4, 3)$

OPEN 表：$(5_2, 2), (6_1, 2), (7_2, 2), (G_3, 2)$

CLOSE 表：$(S, 0), (1_S, 1), (2_S, 1), (3_S, 1), (4_1, 2)$

(6) 挑選：$(5_2, 2)$

子節點：$(2_5, 3), (3_5, 3), (7_5, 3)$

OPEN 表：$(6_1, 2), (7_2, 2), (G_3, 2)$

CLOSE 表：$(S, 0), (1_S, 1), (2_S, 1), (3_S, 1), (4_1, 2), (5_2, 2)$

(7) 挑選：$(6_1, 2)$

子節點：$(1_6, 3)$, $(3_6, 3)$, $(4_6, 3)$, $(7_6, 3), (G_6, 3)$

OPEN 表：$(7_2, 2), (G_3, 2)$

CLOSE 表：$(S, 0), (1_S, 1), (2_S, 1), (3_S, 1), (4_1, 2), (5_2, 2), (6_1, 2)$

(8) 挑選：$(7_2, 2)$

子節點：, $(2_7, 3), (3_7, 3), (5_7, 3), (6_7, 3), (G_7, 3)$

OPEN 表：$(G_3, 2)$

CLOSE 表：$(S, 0), (1_S, 1), (2_S, 1), (3_S, 1), (4_1, 2), (5_2, 2),$

$(6_1, 2), (7_2, 2)$

(9) 挑選：$(G_3, 2)$ =>挑選到終點，則搜尋結束。

(10) 從終點回溯各節點的父節點直到起點為止：$(G_3, 2)$->$(3_S, 1)$-> $(S, 0)$

則搜尋結果之最短路徑：S->3->G

有關 Dijkstra 演算法的詳細搜尋流程，可參考圖 2.4 所示：

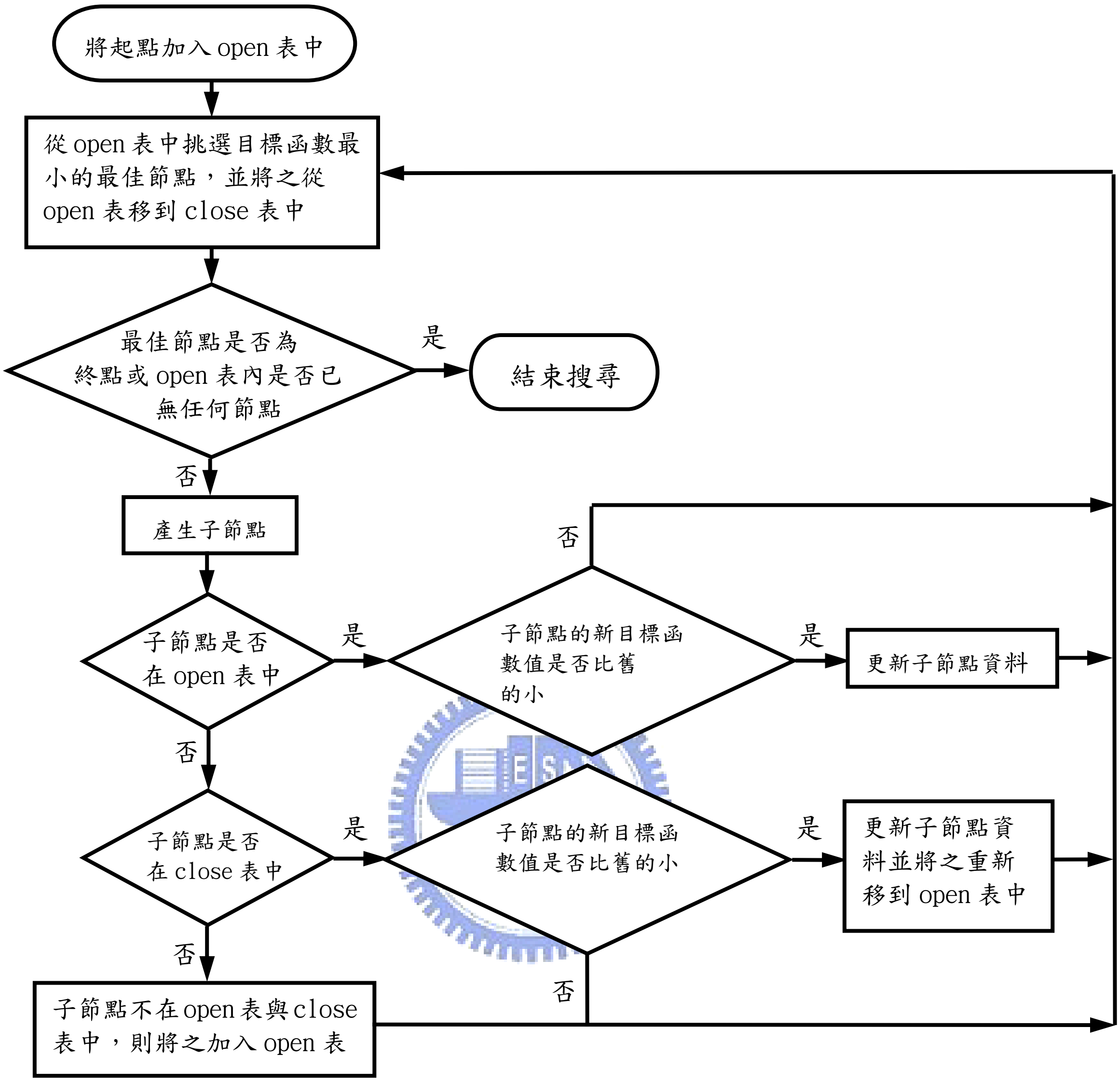

圖 2.4　Dijkstra 演算法流程圖

接著我們以兩個 2D 平面空間最短路徑規劃的例子，來展示 Dijkstra 演算法的成果，如下面圖 2.5 與圖 2.6 所示。其中圖 2.5 為無障礙範例，而圖 2.6 為具障礙物範例。在圖中有顏色的格子為在搜尋中，有被討論到的格子；而顏色越深者表示目標函數值越小、離起點越近，反之則越大。

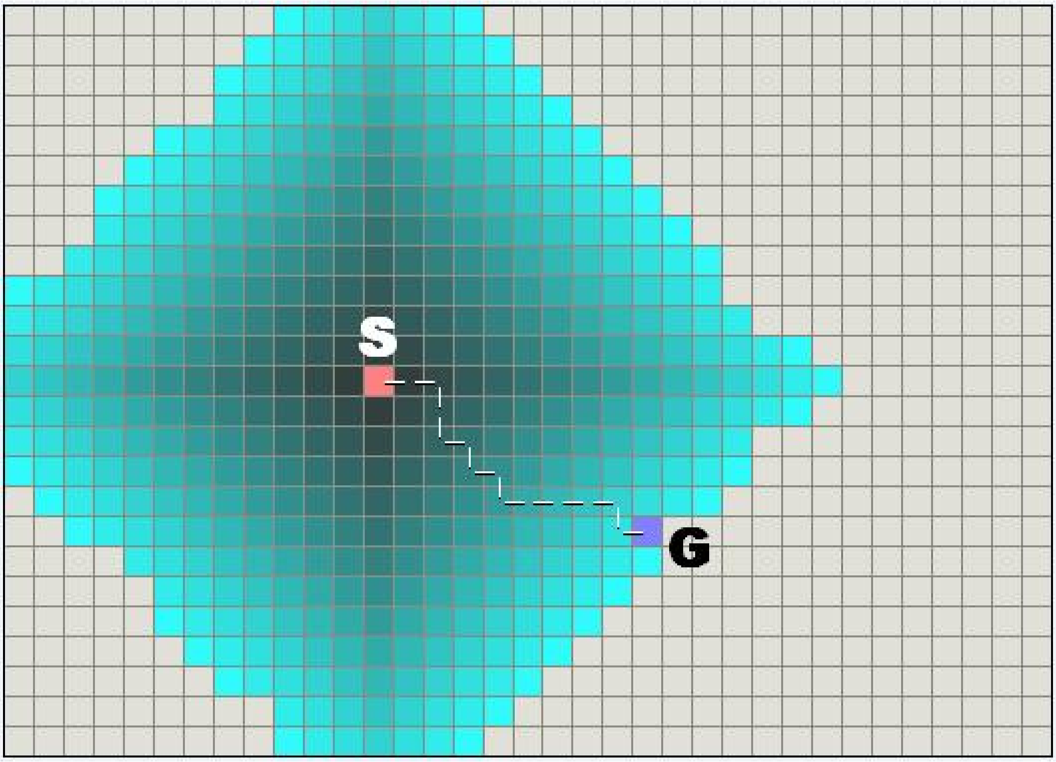

圖 2.5　Dijkstra 演算法無障礙範例
資料來源：參考文獻[8]

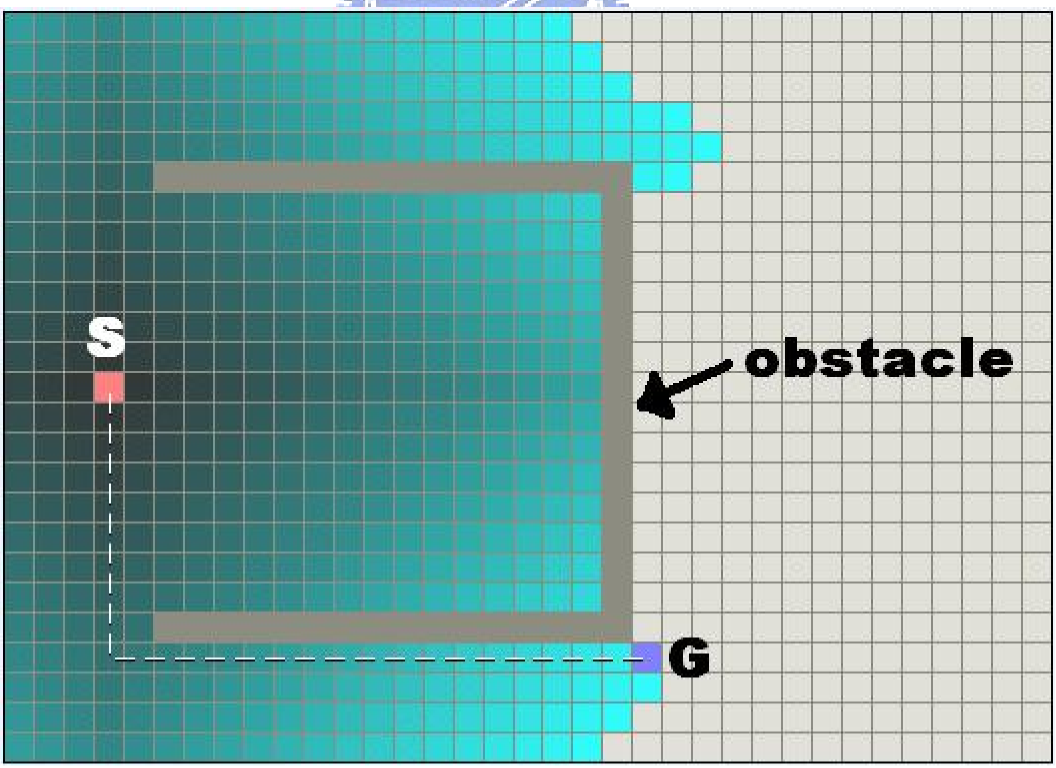

圖 2.6　Dijkstra 演算法具障礙物範例
資料來源：參考文獻[8]

2.3 B-spline Curve

2.3.1 B-spline Curve 的基本理論

1972年，Cox和de Boor[10][11]提出一種新的基底函數 $N_{i,k}(u)$ 來作為曲線的描述基礎。由這種新的基底函數所構成的曲線，我們就稱為B-spline曲線。其遞迴公式如下：

$$P(u)=\sum_{j=0}^{n} P_j N_{i,k}(u) \qquad t_{k-1}\le u\le t_{n+1} \tag{2-38}$$

其中：

$$N_{i,k}(u)=\frac{(u-t_i)N_{i,k-1}(u)}{t_{i+k-1}-t_i}+\frac{(t_{i+k}-u)N_{i+1,k-1}(u)}{t_{i+k}-t_{i+1}} \tag{2-39}$$

$$N_{i,1}(u)=\begin{cases}1 & t_i\le u\le t_{i+1}\\ 0 & otherwise\end{cases} \tag{2-40}$$

在上面三式中，n 為控制點的數目減一，P_j 為控制點的座標，k 為方程式的階數(代表方程式為 $k-1$ 次方)，u 稱為結點(knot)，而 t_i 稱為結點值(knot values)。基底函數 $N_{i,k}(u)$ 的值由 u 與 t_i 的值決定，而 u 的範圍由 t_i 值決定。結點值(t_i)有三種形式：uniform、open uniform 和 nonuniform。在本論文中我們使用 open uniform 的結點值(t_i)，因為 open uniform 的結點值(t_i)具有通過曲線起點與終點的性質，所以非常適合機器手臂路徑規劃的問題。open uniform 的結點值(t_i)由式(2-41)決定：

$$t_i=\begin{cases}0 & 0\le i<k\\ i-k+1 & k\le i\le n\\ n-k+2 & n<i\le n+k\end{cases} \tag{2-41}$$

以下我們將示範一個建構 B-spline Curve 的例子。我們利用三個控制點 P_0,P_1,P_2 來定義一個三階 B-spline 曲線。根據式(2-41)我們先求 t_i 之值：

$$t_0=0,\quad t_1=0,\quad t_2=0,\quad t_3=1,\quad t_4=1,\quad t_5=1$$

接著我們根據式(2-40)求第一階基底函數 $N_{i,1}(u)$ 的值：

$$N_{0,1}(u)=\begin{cases}1 & t_0 \le u \le t_1 \\ 0 & otherwise\end{cases} \quad (u=0)$$

$$N_{1,1}(u)=\begin{cases}1 & t_1 \le u \le t_2 \\ 0 & otherwise\end{cases} \quad (u=0)$$

$$N_{2,1}(u)=\begin{cases}1 & t_2 \le u \le t_3 \\ 0 & otherwise\end{cases} \quad (u\le 1)$$

$$N_{3,1}(u)=\begin{cases}1 & t_3 \le u \le t_4 \\ 0 & otherwise\end{cases} \quad (u=1)$$

$$N_{4,1}(u)=\begin{cases}1 & t_4 \le u \le t_5 \\ 0 & otherwise\end{cases} \quad (u=1)$$

對於第一階基底函數值的選擇中，我們令當 $t_i=t_{i+1}$ 時，$N_{i,1}(u)=0$。所以在此例中，只有 $N_{2,1}(u)$ 在 $t_2\le u\le t_3$ 時，其值為 1，其他第一階基底函數的值永遠為零。當所有的第一階基底函數 $N_{i,1}(u)$ 都求出之後，便可以求下一階的基底函數 $N_{i,2}(u)$，以此類推。而從第二階以後的基底函數就必須根據式(2.39)求出，所以第二階的基底函數分別為：

$$N_{1,2}(u)=\frac{(u-t_1)N_{1,1}}{t_2-t_1}+\frac{(t_3-u)N_{2,1}}{t_3-t_2}=(1-u)$$

$$N_{2,2}(u)=\frac{(u-t_2)N_{2,1}}{t_3-t_2}+\frac{(t_4-u)N_{3,1}}{t_4-t_3}=u$$

同樣地，我們可以得到第三階基底函數 $N_{i,3}(u)$：

$$N_{0,3}(u)=\frac{(u-t_0)N_{0,2}}{t_3-t_1}+\frac{(t_3-u)N_{1,2}}{t_4-t_2}=(1-u)^2$$

$$N_{1,3}(u)=\frac{(u-t_1)N_{1,2}}{t_3-t_1}+\frac{(t_4-u)N_{2,2}}{t_4-t_2}=2u(1-u)$$

$$N_{2,3}(u)=\frac{(u-t_2)N_{2,2}}{t_4-t_2}+\frac{(t_5-u)N_{3,2}}{t_5-t_2}=u^2$$

由於我們要的是三階 B-spline 曲線，所以當第三階基底函數 $N_{i,3}(u)$ 求出後，我們就可以將 $N_{i,3}(u)$ 與控制點(P_0,P_1,P_2)代入式(2-38)中來得到我們要的曲線方程式。

$$P(u)=(1-u)^2P_0+2u(1-u)P_1+u^2P_2$$

以下我們實際撰寫一個程式(如圖 2.7)來驗證 B-spline Curve 理論。

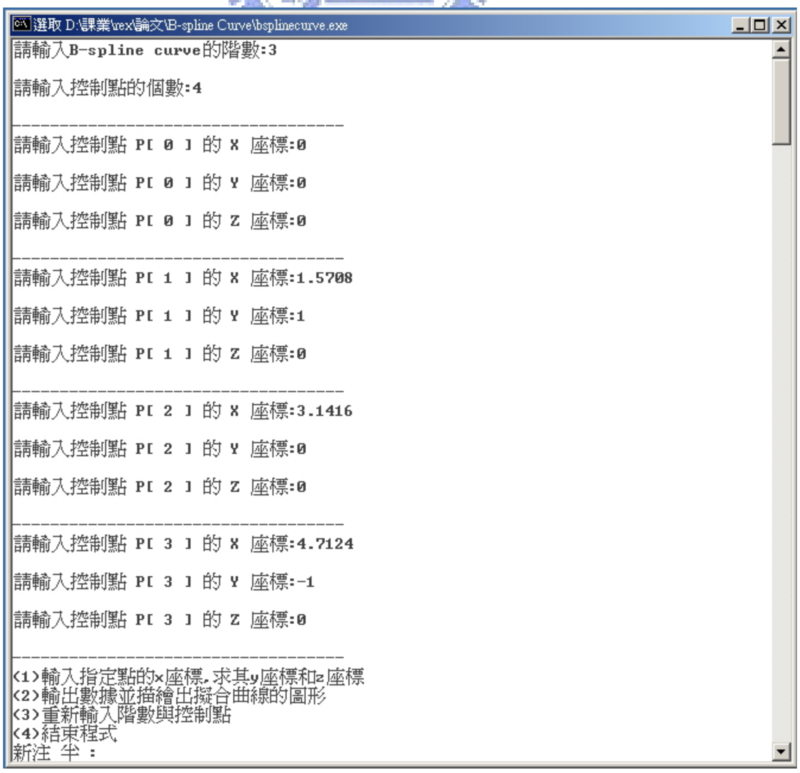

圖 2.7 B-spline Curve 程式執行畫面

下面將示範兩個範例，範例一是以五個控制點來擬合 $y=\sin x$，$0 \le x \le \pi$，如圖 2.8。而範例二是以十個控制點來擬合 $y=\cos x$，$0 \le x \le \pi$，如圖 2.9。由這兩個圖比較結果來看，在相同的範圍內控制點越多，則曲線擬合的效果越好。(範例說明：綠線部份為以三階 B-spline Curve 擬合之曲線；藍線部份為以四階 B-spline Curve 擬合之曲線；紅線部份為實際之曲線。)

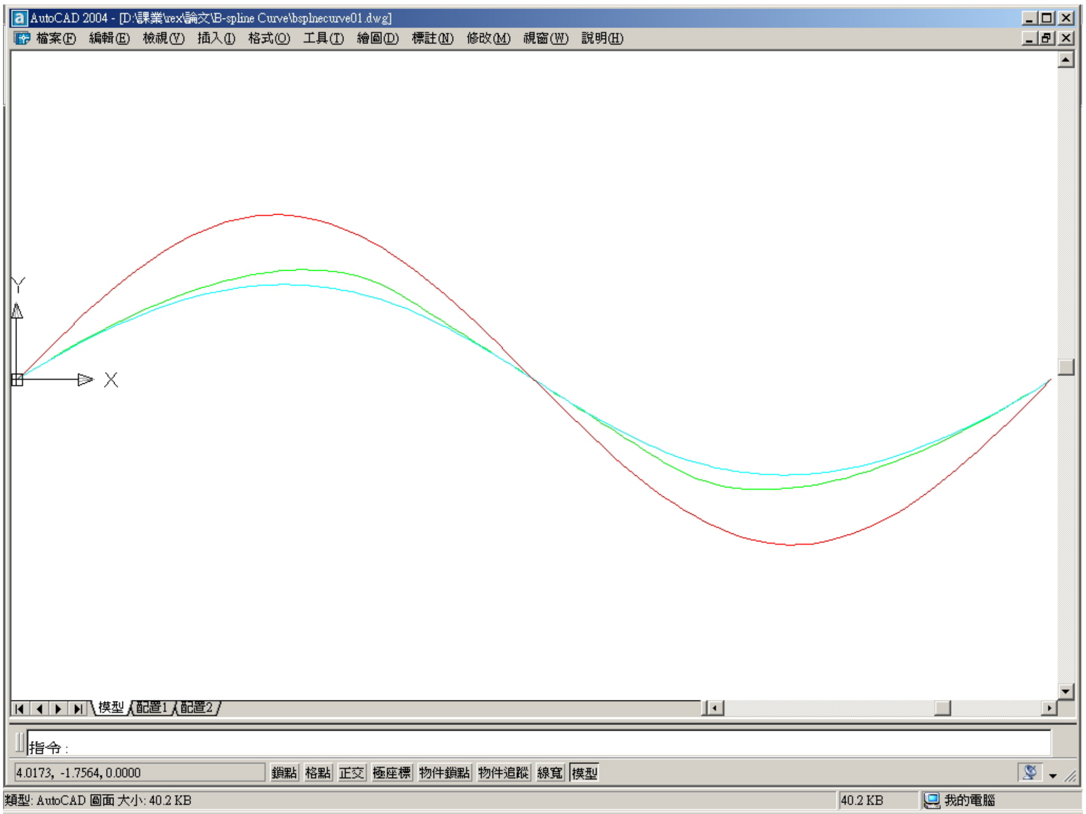

圖 2.8　　B-spline Curve 之範例一(以五個控制點擬合 y=sinx)

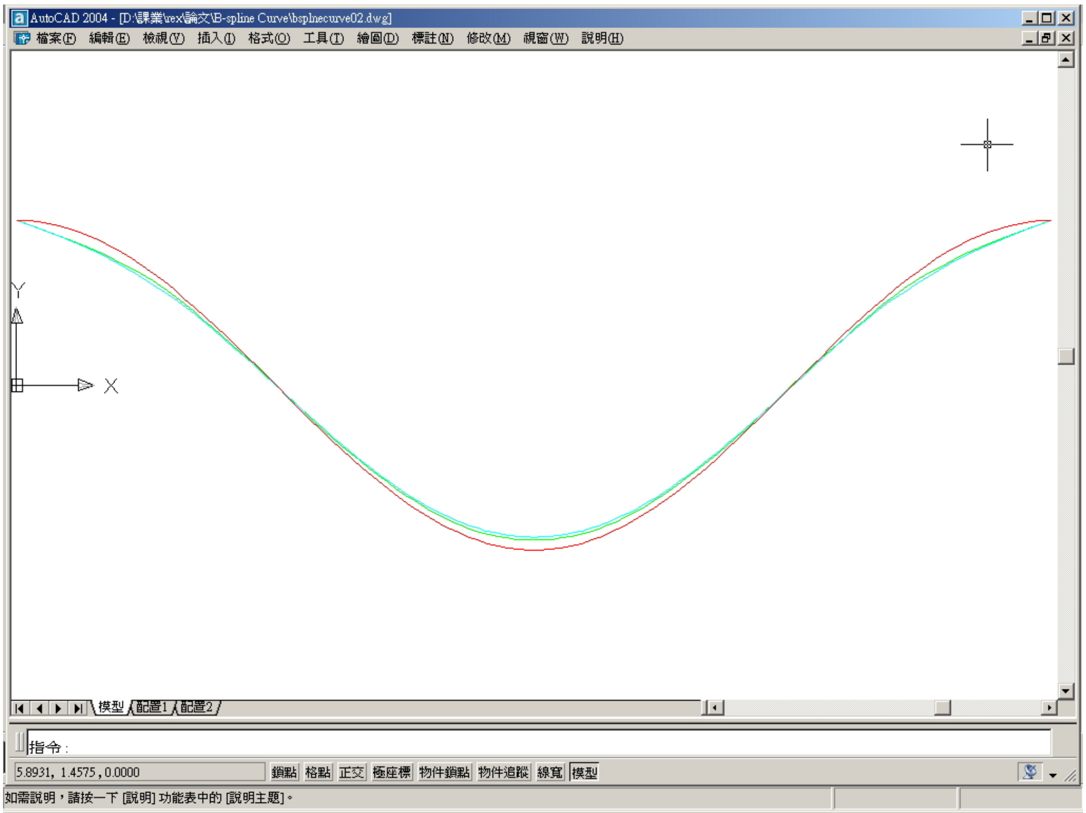

圖 2.9　　B-spline Curve 之範例二(以十個控制點擬合 y=cosx)

2.3.2 B-spline Curve Fit

對於B-spline曲線，最大的缺點就是曲線不會通過控制點，即使是open uniform 的 B-spline 曲線也只會通過起點與終點。為了能使我們建構出來的 B-spline 曲線都能通過控制點，我們必須找出另一組新控制點群，使由新控制點群所建構出來的 B-spline 曲線都能通過控制點。下面式(2-42)中，$P_j(u_j)$ 表示 B-spline 曲線欲通過之控制點群，B_n 表示建構出來的 B-spline 曲線都能通過控制點的新控制點群，$N_{n,k}$ 表示 B-spline 曲線的基底函數，n 為新控制點群的個數，k 為 B-spline 曲線的階數。

$$P_1(u_1) = N_{1,k}(u_1) \bullet B_1 + N_{2,k}(u_1) \bullet B_2 + \ldots + N_{n,k}(u_1) \bullet B_n$$

$$P_2(u_2) = N_{1,k}(u_2)\bullet B_1 + N_{2,k}(u_2)\bullet B_2 + \ldots + N_{n,k}(u_2)\bullet B_n$$

$$\bullet$$
$$\bullet$$
$$\bullet$$

$$P_j(u_j) = N_{1,k}(u_j)\bullet B_1 + N_{2,k}(u_j)\bullet B_2 + \ldots + N_{n,k}(u_j)\bullet B_n$$

$$where \quad 2 < k < n \le j \tag{2-42}$$

我們將式(2-42)簡化成矩陣可表為下面式(2-43)：

$$[P]_{j\times 1} = [N]_{j\times n}[B]_{n\times 1} \tag{2-43}$$

在式(2-43)中，如果 j=n，則表示矩陣$[N]$為一方陣。若矩陣$[N]$為一方陣，且$[N]$的行列式值不為零，則$[N]$具有反矩陣$[N]^{-1}$；我們欲求新控制點群之矩陣$[B]$，則將式(2-43)等式兩邊同乘$[N]^{-1}$，即可得：

$$[N]^{-1}{}_{n\times n}[P]_{n\times 1} = [B]_{n\times 1} \tag{2-44}$$

如此我們就可以得到一組新控制點群$[B]$，使由新控制點群所建構出來的 B-spline 曲線都能通過控制點群$[P]$。倘若我們欲減少新控制點群的個數，使得 n<j，則矩陣$[N]$便不為一方陣，也不具反矩陣，這麼一來我們就不能利用式(2-44)來求得新控制點群$[B]$。但若我們先將式(2-43)等式兩邊同乘$[N]$的轉置矩陣$[N]^T$，我們可以將式(2-43)改為：

$$\begin{aligned} &[N]^T{}_{n\times j}[P]_{j\times 1} = [N]^T{}_{n\times j}[N]_{j\times n}[B]_{n\times 1} \\ \rightarrow &[N^T P]_{n\times 1} = [N^T N]_{n\times n}[B]_{n\times 1} \end{aligned} \tag{2-45}$$

如此$[N^T N]$即為一方陣，若$[N^T N]$的行列式值不為零，則$[N^T N]$具有反矩陣$[N^T N]^{-1}$。我們欲求新控制點群之矩陣$[B]$，則將式(2-45)等式兩邊同乘

$[N^T N]^{-1}$，即可得：

$$[N^T N]^{-1}{}_{n\times n}[N^T P]_{n\times 1} = [B]_{n\times 1} \tag{2-46}$$

對於式(2-42)中的 u 值該如何決定，我們採用的方法為以原控制點間的直線距離之比例來決定：

$$\begin{aligned} & u_1 = 0 \\ & \frac{u_i}{u_{\max}} = \frac{\sum_{s=2}^{i} |P_s - P_{s-1}|}{\sum_{s=2}^{j} |P_s - P_{s-1}|} \end{aligned} \tag{2-47}$$

其中 $u_{\max}$ 的值則等於式(2-41)中的結點最大值 $t_{\max}$。

2.4 OpenGL 函式庫

OpenGL[12]是”Open Graphics Library”的縮寫，顧名思義它是開放的繪圖函式庫，只要不從事任何有關商業行為，它是可以被任意修改與使用的。基本上，它是 3D 繪圖與模型程式庫，即一種 3D API。OpenGL 不像是 C 或 C++那樣的程式語言，它比較像是 C 執行時期的程式庫(runtime library)。OpenGL 是與硬體無關的軟體介面，可以在不同的平臺如 Windows、Unix、Linux、MacOS、OS／2 之間進行移植。因此，支援 OpenGL 的軟體具有很好的移植性，可以獲得非常廣泛的應用。OpenGL 使用了由 Silicon Graphics 公司所研發且最佳化的演算法，該公司是世界上電腦繪圖與動畫領域的領導者。OpenGL 有許多用途，從 CAD 工程與建築應用到熱門電影中擬真的動畫繪製。雖然 MicroSoft 的 DirectX 在遊戲類市場全面

領先，但在專業高級繪圖領域，OpenGL 依然是不能被取代的主角。

在本論文中，我們使用 DEV C++來從事模擬程式的撰寫。DEV C++目前為免費使用的 C/C++編譯軟體，它具有多平台、多語言的版本，使用者可以免費從官方網頁下載最新版本。另外，我們所使用的電腦軟硬體規格，整理成下面表 2.2 所示，以供比較參考：

表 2.2 本論文所使用的電腦軟硬體規格

硬體名稱	**規格**
主機板	MSI KT2 Combo (MS-6764)
CPU	AMD Duron XP, 1200 MHz
記憶體	DDR400 512MB
顯示卡	Asus V3800M TV Video Adapter 32MB
作業系統	Microsoft Windows 2000 Professional
OpenGL 版本	1.3.1