


## 第五章、以動力學討論三軸機器手臂之路徑規劃問題

在本章中，我們將綜合第三章與第四章所發展的方法和理論，並以動力學來討論三軸機器手臂之路徑規劃問題。在 5.1 節中，我們先以動力學的觀念與 B-spline Curve 的理論來提出新的目標函數。在 5.2 節中，我們使用第四章提出的 Dijkstra 演算法加網格漸細法來以動力學討論路徑規劃的問題。

### 5.1 動力問題的目標函數

#### 5.1.1 旋轉運動學



對於本論文中所討論的三軸機器手臂，其桿件間的運動均屬簡單的旋轉運動。在旋轉運動學中，常以旋轉角度  $\theta$  表示位置的變化， $\theta$  對時間的變化率稱為角速度  $\dot{\theta}$ ，角速度對時間的變化率則稱為角加速度  $\ddot{\theta}$ 。如圖 5.1 為一旋轉角度對時間的關係圖，當角度變化為由  $\theta \rightarrow \theta'$ ，時間變化為  $\Delta t$ ，則由  $\theta \rightarrow \theta'$  的平均角速度為：

$$\dot{\theta}_{avg} = \frac{\theta' - \theta}{\Delta t} \quad (5-1)$$

當  $\Delta t$  極小甚至趨近於零的時候，則平均角速度  $\dot{\theta}_{avg}$  則趨近於位於  $\theta'$  時的瞬間角速度：

$$\dot{\theta} = \lim_{\Delta t \rightarrow 0} \frac{\theta' - \theta}{\Delta t} = \frac{d\theta}{dt} \quad (5-2)$$

如圖 5.2 為一角速度對時間變化的關係圖，當角速度變化為由  $\dot{\theta} \rightarrow \dot{\theta}'$ ，時間變化為  $\Delta t$ ，則由  $\dot{\theta} \rightarrow \dot{\theta}'$  的平均角加速度為：

$$\ddot{\theta}_{avg} = \frac{\dot{\theta}' - \dot{\theta}}{\Delta t} \quad (5-3)$$

當  $\Delta t$  極小甚至趨近於零的時候，則平均角加速度  $\ddot{\theta}_{avg}$  則趨近於位於  $\dot{\theta}'$  時的瞬間角加速度：

$$\ddot{\theta} = \lim_{\Delta t \rightarrow 0} \frac{\dot{\theta}' - \dot{\theta}}{\Delta t} = \frac{d\dot{\theta}}{dt} \quad (5-4)$$

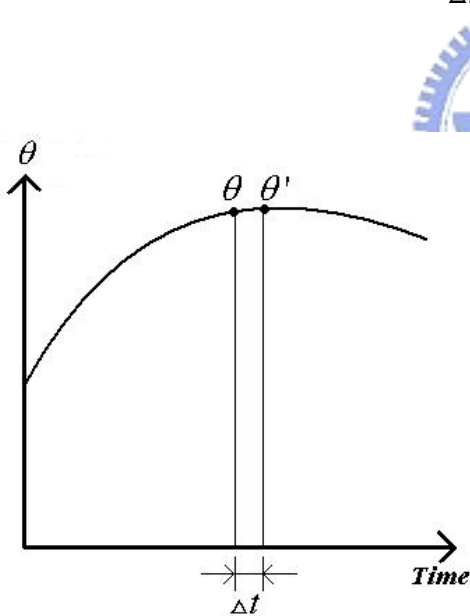


圖 5.1 角度對時間變化的關係圖

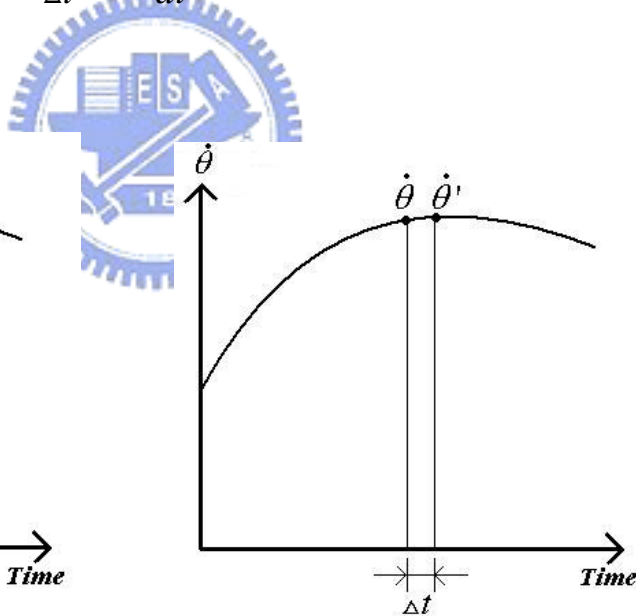


圖 5.2 角速度對時間變化的關係圖

### 5.1.2 B-spline 參數化路徑

如上一小節所述，若我們欲計算瞬間角速度與角加速度，我們必須先知道微小角度變化量與時間的關係。但是因為我們建立問題模型空間是以

網格方式建立許多節點資料。當節點間的距離過大，則無法計算瞬間角速度與角加速度；若我們節點間的距離極小，雖然可以計算瞬間角速度與角加速度，但搜尋空間變極大，搜尋的時間將變得非常久，這也不是我們所該採取的方法。所以我們提出透過 B-spline Curve 的理論，來參數化節點間的路徑，這樣我們可以藉著參數化後的路徑來計算瞬間角速度與角加速度。對於具有時間參數的節點路徑，我們依然可以利用 B-spline Curve 求得參數化路徑，並且包含時間參數。如下面式(5-5)：

$$\begin{bmatrix} P_{\theta_1}(u) \\ P_{\theta_2}(u) \\ P_{\theta_3}(u) \\ P_t(u) \end{bmatrix} = \sum_{j=0}^n N_{i,k}(u) \cdot \begin{bmatrix} P_{\theta_1,j} \\ P_{\theta_2,j} \\ P_{\theta_3,j} \\ P_{t,j} \end{bmatrix} \quad (5-5)$$

其中  $P_{\theta_1}(u)$ 、 $P_{\theta_2}(u)$ 、 $P_{\theta_3}(u)$  為 B-spline Curve 路徑上節點的三軸角度向量， $P_t(u)$  為 B-spline Curve 路徑上節點的時間向量， $P_{\theta_1,j}$ 、 $P_{\theta_2,j}$ 、 $P_{\theta_3,j}$ 、 $P_{t,j}$  為控制點，即為原結果路徑上節點的三軸角度與時間向量， $n$  為控制點數目，即為原結果路徑的節點數目。根據 B-spline Curve 的理論，要建構一條 B-spline Curve，至少需要三個控制點。所以我們取節點、父節點與父節點之父節點當作控制點，來建構這三點間的 B-spline Curve 路徑，如圖 5.3 所示。雖然我們可以 B-spline Curve 的理論來參數化節點間的路徑，但路徑僅會通過曲線的起點( $N_{i-2}$ )與終點( $N_i$ )。這樣當節點間的距離很大的時候，計算位於  $N_{i-1}$  附近的角速度與角加速度的誤差會很大。為了避免上述的

情況發生，我們使用 2.3.2 節的 B-spline Curve Fit 的理論，來使新控制點所建構出來的 B-spline Curve 均會通過搜尋節點，如圖 5.4、圖 5.5 所示。

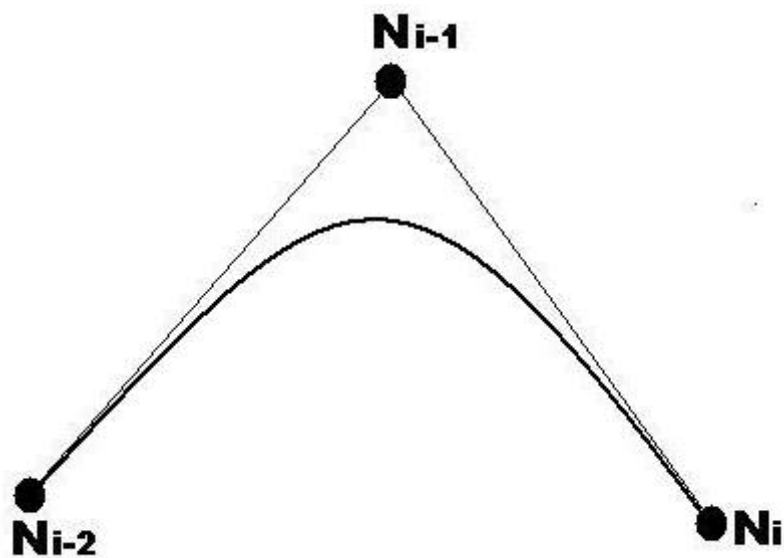


圖 5.3 以三點建構的 B-spline Curve

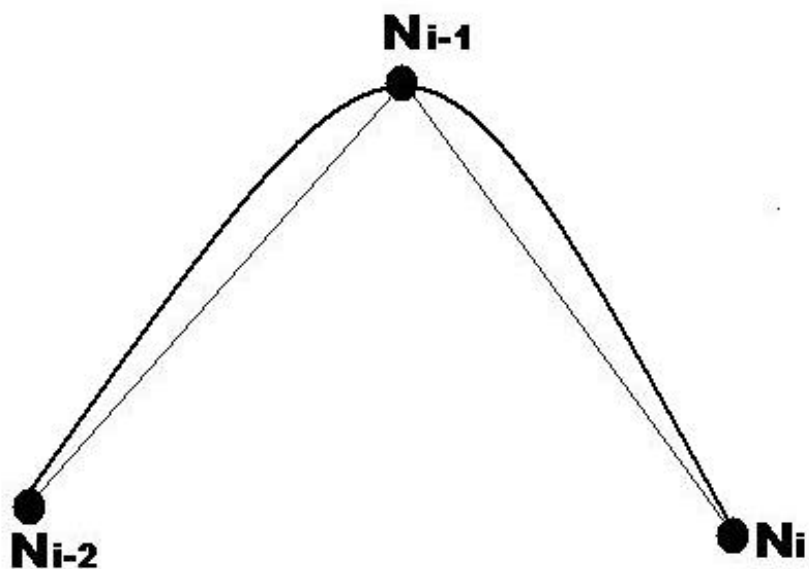


圖 5.4 有做 curve fit 動作的 B-spline Curve

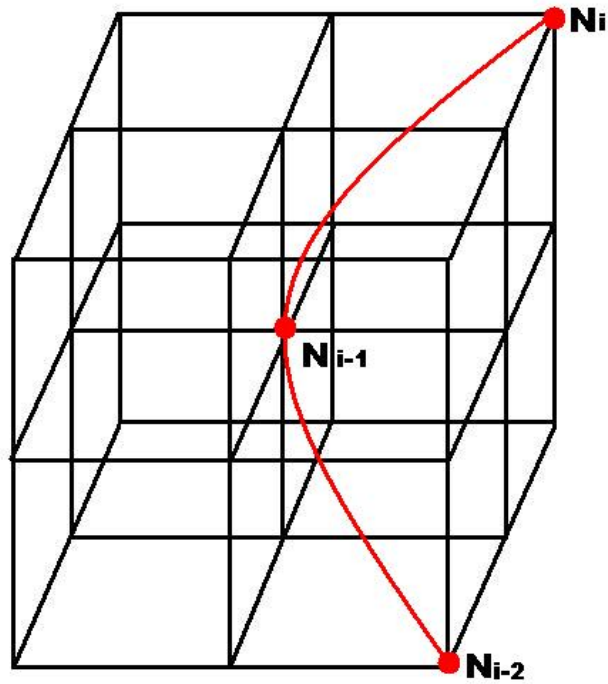


圖 5.5 有做 curve fit 動作的 B-spline Curve(三維空間)

### 5.1.3 目標函數的設計

式(3-2)為計算驅動器所承受力矩的動力方程式。其中  $M(\ddot{\theta})\dot{\theta}$  為慣性力矩， $V(\theta, \dot{\theta})$  為科氏力矩與離心力矩，而  $G(\theta)$  為重力力矩。由方程式的結構我們可以看出，力矩的大小變化與時間有很大的關係。

$$\tau = M(\ddot{\theta})\dot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \quad (5-6)$$

在相同的角度變化量下，若經過的時間越短，則瞬間角速度與角加速度就越大，相對地瞬間力矩也會越大。然而若經過的時間很久，固然瞬間力矩會比較小；但長時間受到力矩的作用，其所受的力矩總和卻不一定會比短時間的小。因此我們要找到最佳的時間點，來平衡瞬間力矩與力矩總和的付出成本。如圖 5.6 所示，為兩節點間移動時的力矩對時間的變化示意圖。

若兩節點間的力矩呈非線性連續變化，則兩節點間移動時的力矩總和為力矩對時間的積分。為了求出節點間的力矩對時間的積分，我們使用數值積分法的梯形法則(trapezoid rule)來求積分的近似值。如圖 5.6 中，我們在節點  $N_{i-1}$  與  $N_i$  間取  $j$  個點 ( $P_j$ )，這些點(包括節點)兩兩可構成一個梯形，則節點  $N_{i-1}$  到  $N_i$  的力矩對時間的積分可以利用這  $j+1$  個梯形面積來近似。

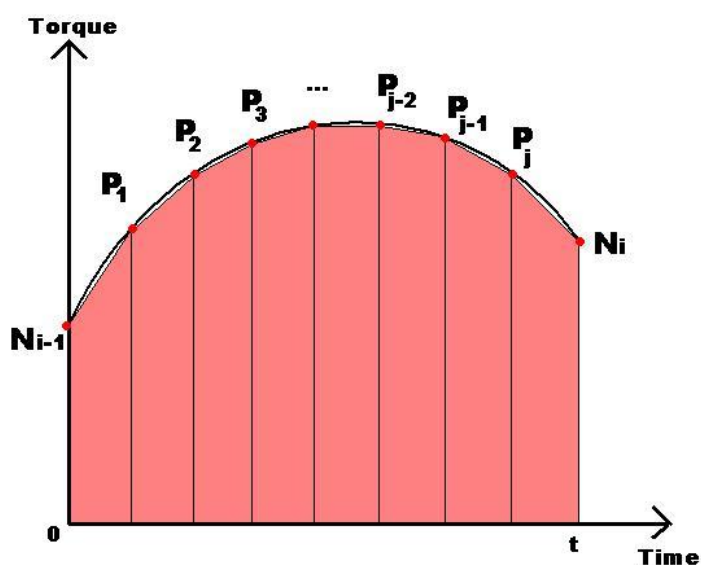
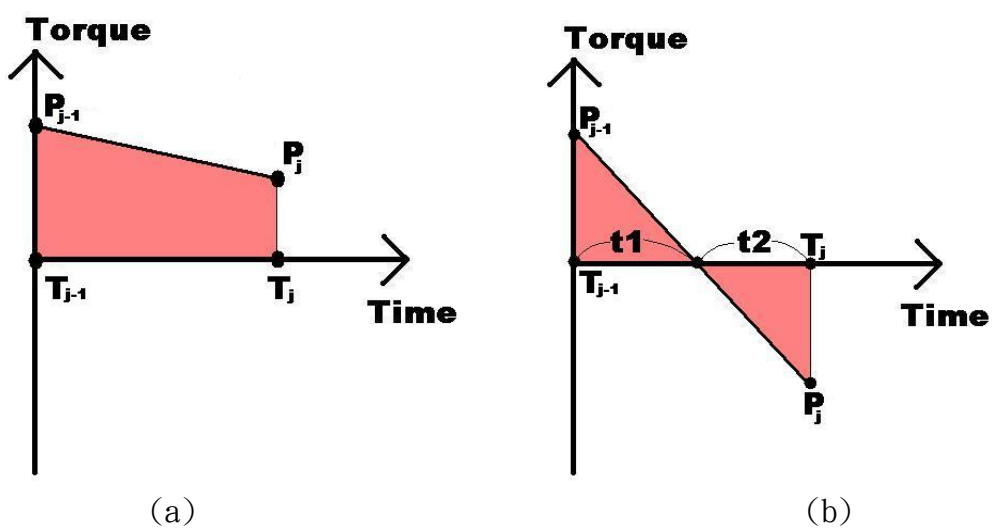


圖 5.6 節點間的力矩變化示意圖(一)



(a)

(b)

圖 5.7 節點間的力矩變化示意圖(二)

(a)兩節點的力矩為同號 (b) 兩節點的力矩為異號

如圖 5.7，當兩節點的力矩為同號時，兩節點力矩對時間變化的近似圖形為梯形，則其面積  $A_j$  計算如式(5-7)：

$$A_j = \frac{(|P_j| + |P_{j-1}|) \times (T_j - T_{j-1})}{2} \quad (5-7)$$

而當兩節點的力矩為異號時，則兩節點力矩對時間變化的近似圖形為兩個三角形，則其面積  $A_j$  計算如式(5-8)：

$$A_j = \frac{|P_{j-1}| \times t_1}{2} + \frac{|P_j| \times t_2}{2} \quad (5-8)$$

其中  $t_1$ 、 $t_2$  根據相似三角形理論計算如式(5-9)、(5-10)：

$$t_1 = (T_j - T_{j-1}) \times \frac{|P_{j-1}|}{|P_j| + |P_{j-1}|} \quad (5-9)$$

$$t_2 = (T_j - T_{j-1}) \times \frac{|P_j|}{|P_j| + |P_{j-1}|} \quad (5-10)$$

所以我們設計節點  $N_{i-1}$  到  $N_i$  的力矩付出總成本為其力矩的絕對值對時間的積分；欲求力矩付出成本最小的情況，即求  $j+1$  個梯形面積和最小的情況，如式(5-11)所示：

$$\text{Minimize } H = \sum_{j=1}^m A_j \quad (5-11)$$

而目標函數的設計，則將所有的力矩付出成本  $H$  給加總並取其最小值。如式(5-12)所示：

$$\text{Minimize } f = \sum_{i=1}^n \left( |H_1|^\omega + |H_2|^\omega + |H_3|^\omega \right) \quad (5-12)$$

其中  $H_1$ 、 $H_2$ 、 $H_3$  分別表示兩節點間三軸各自所承受的力矩付出成本， $n$  為

最後路徑的總節點數， $\omega$  為次方係數。在圖 5.6 中， $N_{i-1}$  到  $N_i$  間的  $j+2$  個點的力矩值，是根據式(2-34)、(2-36)、(2-37)來計算之。至於每個點瞬間的角速度與角加速度，我們以參數化的路徑來求得。如圖 5.8，欲求在  $P_j$  點的角速度與角加速度，我們在路徑上往前取兩個非常接近  $P_j$  的兩個點  $P_j'$  與  $P_j''$ 。則  $P_j$  點的角速度與角加速度計算式如下：

$${}^{P'_j} \dot{\theta}_k = \frac{{}^{P'_j} \theta_k - {}^{P''_j} \theta_k}{{}^{P'_j} t - {}^{P''_j} t} \quad (5-13)$$

$${}^{P_j} \dot{\theta}_k = \frac{{}^{P_j} \theta_k - {}^{P'_j} \theta_k}{{}^{P_j} t - {}^{P'_j} t} \quad (5-14)$$

$${}^{P_j} \ddot{\theta}_k = \frac{{}^{P_j} \dot{\theta}_k - {}^{P'_j} \dot{\theta}_k}{{}^{P_j} t - {}^{P'_j} t} \quad (5-15)$$

where  $k=1,2,3$

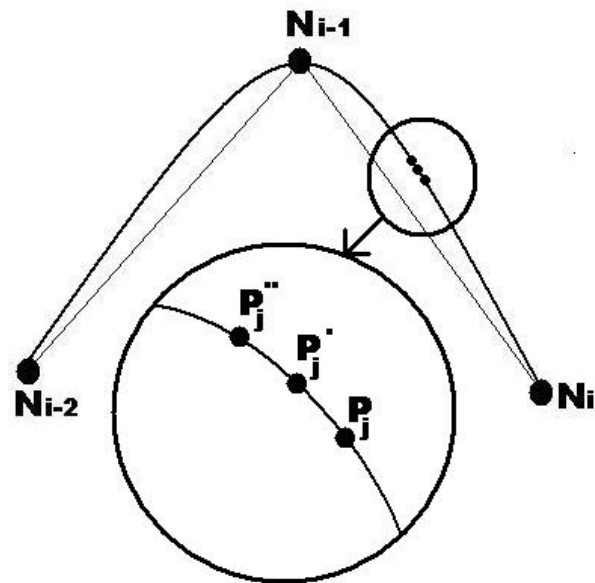


圖 5.8 參數化後的路徑上極接近的三個點



## 5.2 路徑搜尋與結果

對於動力問題來說，時間變為很重要的參數。所以現在我們紀錄節點資料有：(1)節點的位置 $(\theta_1, \theta_2, \theta_3)$ (2)節點的父節點位置(3)節點的目標函數值(4)從父節點到節點的時間。父節點到節點的時間，我們設定了從 0.001 秒到 10 秒間作搜尋。其中 0.001 秒到 0.100 秒取 100 個時間點、0.10 到 1.00 取 100 個點、1.0 到 10.0 取 100 個點，總共約 300 個時間點作計算。從這個範圍內決定節點間最佳的移動時間點。在本文所謂的最佳的移動時間，為父節點到子節點的力矩對時間積分最小。其積分的計算方式則用上一節所述的梯形法則作近似值運算。接著我們可以針對動力問題來規劃路徑。

我們以(5-12)式為目標函數，且加入與第四章相同的限制條件：(1) $\theta_1$ 、 $\theta_2$ 與 $\theta_3$ 的範圍必須在正負 150 度之間 (2)每一軸所能承受的力矩大小在正負 15 Nm 之間。搜尋方法則採用第四章所發展的 Dijkstra 演算法加網格漸細法。我們首先以步階=20 度作第一次粗略路徑搜尋，接著再以步階=10 度作第二次路徑搜尋，最後以步階=5 度作第三次路徑搜尋；並設定 $\omega$ 為 1。其搜尋路徑與動態模擬的結果如圖 5.9~圖 5.11 所示：

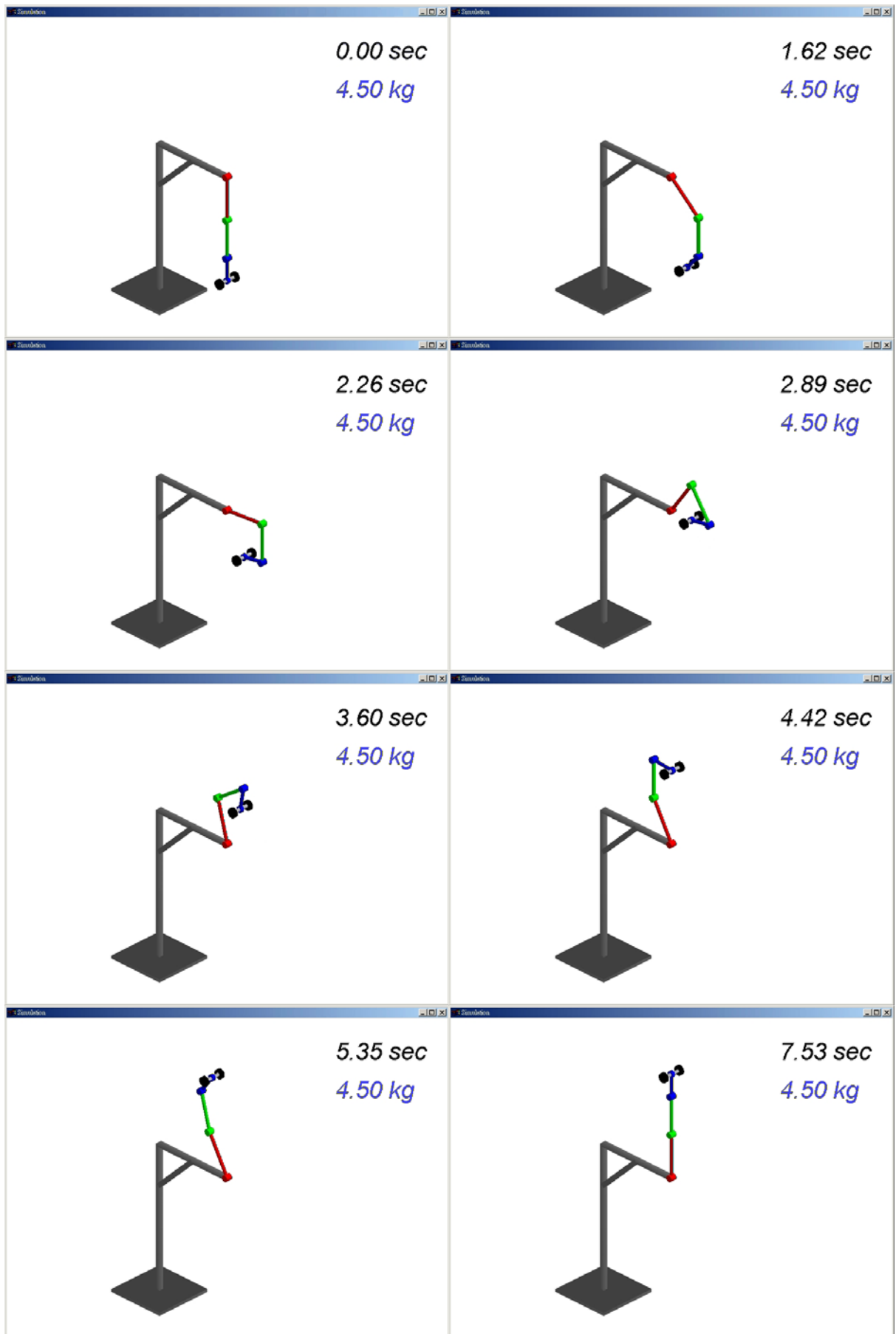


圖 5.9 動態模擬結果(網格漸細法，次方係數為1)

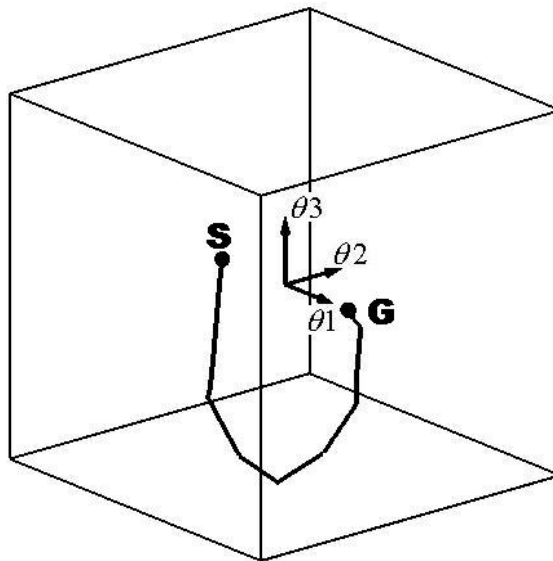


圖 5.10 模擬結果之行進路徑(網格漸細法，次方係數為1)

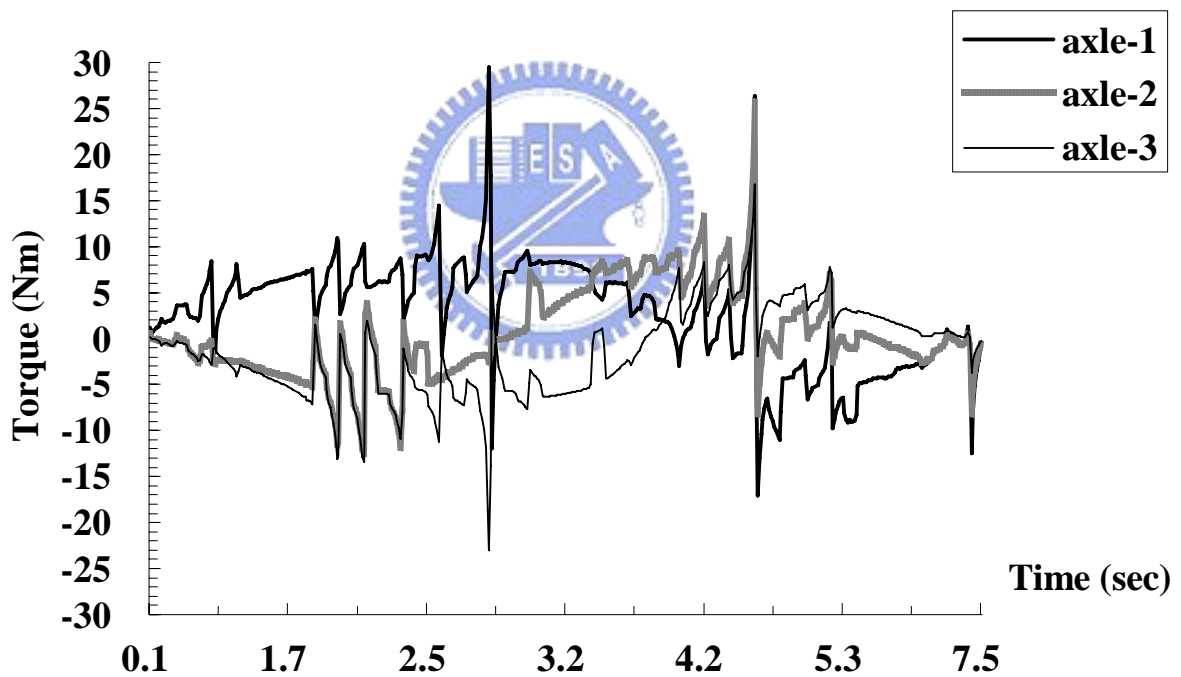


圖 5.11 力矩與舉重時間的關係圖(網格漸細法，次方係數為1)

接著我們將(5-12)式中的次方係數 $\omega$ 設定為2產生新的目標函數，其它設定與上一個例子相同。其搜尋路徑與動態模擬的結果如圖 5.12~圖 5.14 所示：

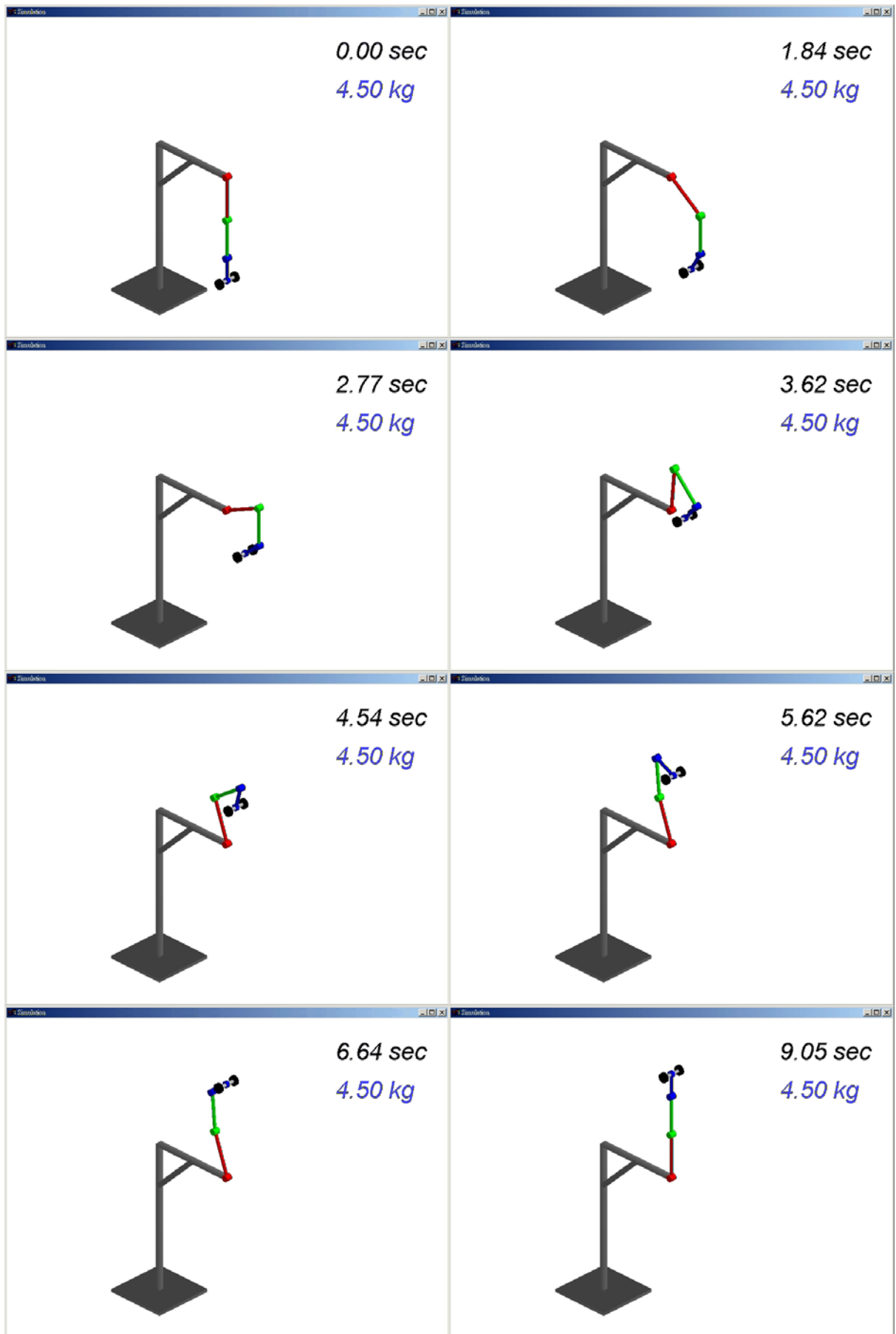


圖 5.12 動態模擬結果(網格漸細法，次方係數為 2)

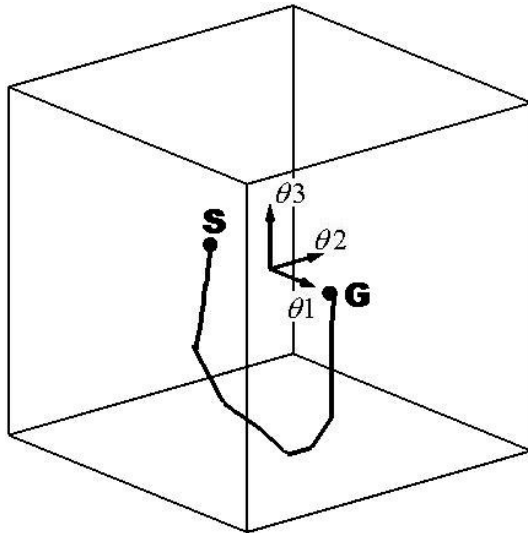


圖 5.13 模擬結果之行進路徑(網格漸細法，次方係數為 2)

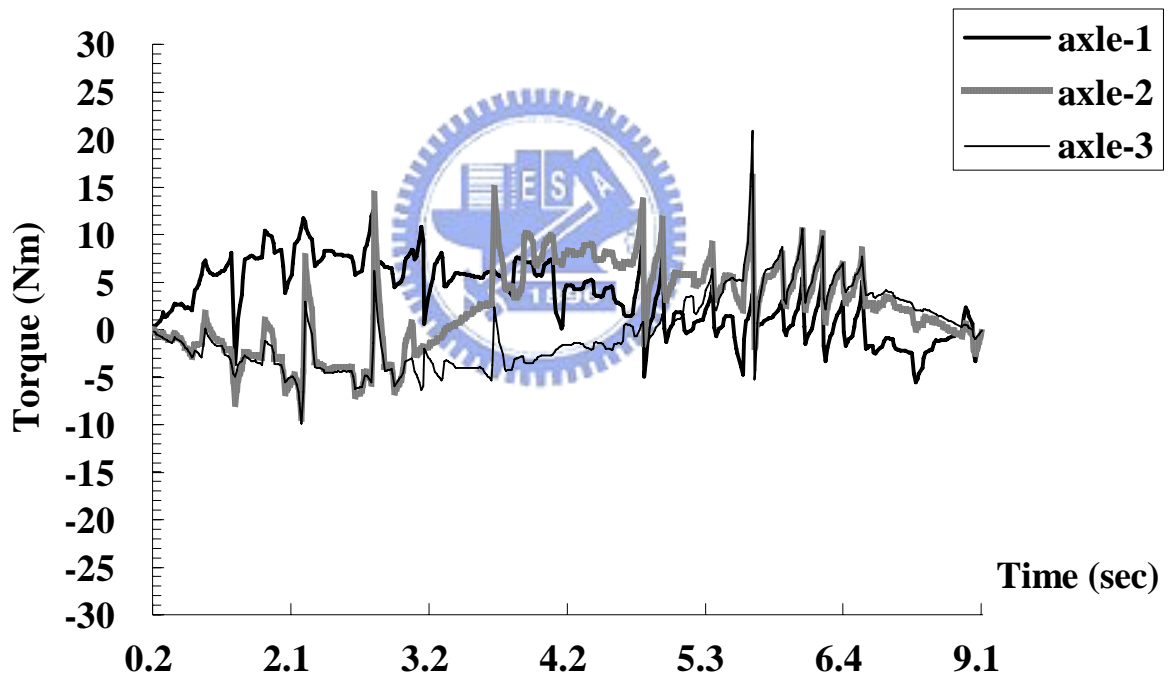


圖 5.14 力矩與舉重時間的關係圖(網格漸細法，次方係數為 2)

由上面兩個例子的結果我們可以看出，提高次方係數 $\omega$ 由 1 至 2，可以有效地平均三軸瞬間力矩的大小，並且降低各軸所承受之最大力矩負荷與減少力矩超過限制條件的情形。

### 5.3 與文獻結果的比較

在本節中，我們將機器手臂的尺寸改為文獻[4][5](如圖 5.15)。其限制條件為：(1) $\theta_1$ 、 $\theta_2$ 與 $\theta_3$ 的範圍必須在正負 150 度之間 (2)每一軸所能承受的力矩大小在正負 50 Nm 之間。我們將用本論文的方法所搜尋的結果與文獻[4][5]的結果(圖 5.19)來作比較並討論。搜尋方法同樣以 Dijkstra 演算法加網格漸細法搜尋，以步階=20 度作第一次粗略路徑搜尋，接著再以步階=10 度作第二次路徑搜尋，最後以步階=5 度作第三次路徑搜尋；並設定次方係數 $\omega$ 為 2，其搜尋結果如圖 5.16~圖 5.18 所示：

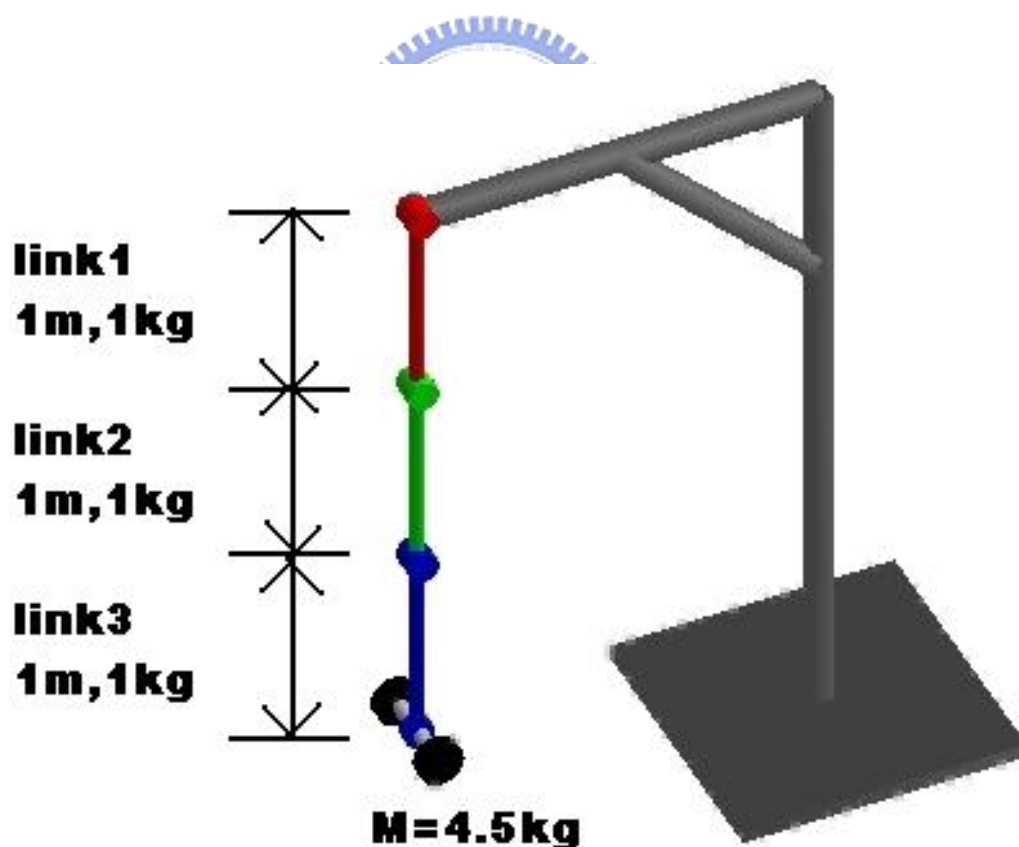


圖 5.15 文獻[4][5]所討論的機器手臂

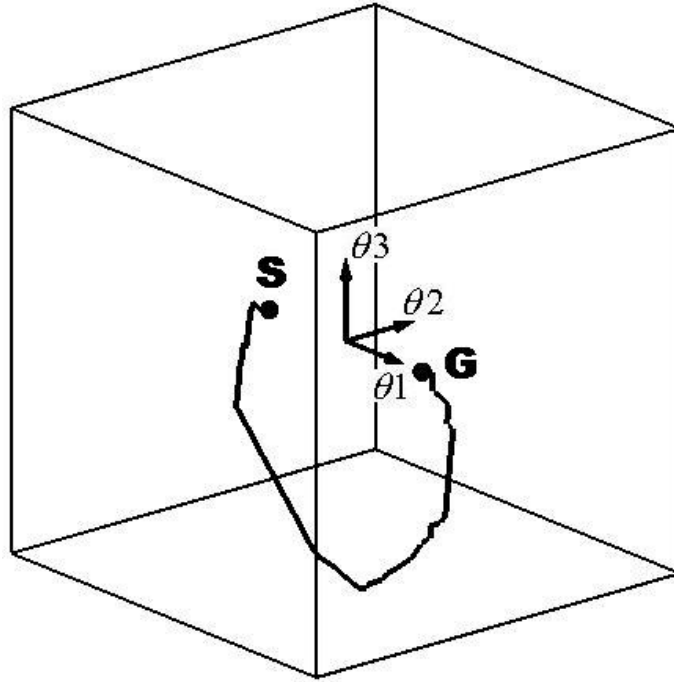


圖 5.16 模擬結果之行進路徑(次方係數為 2)

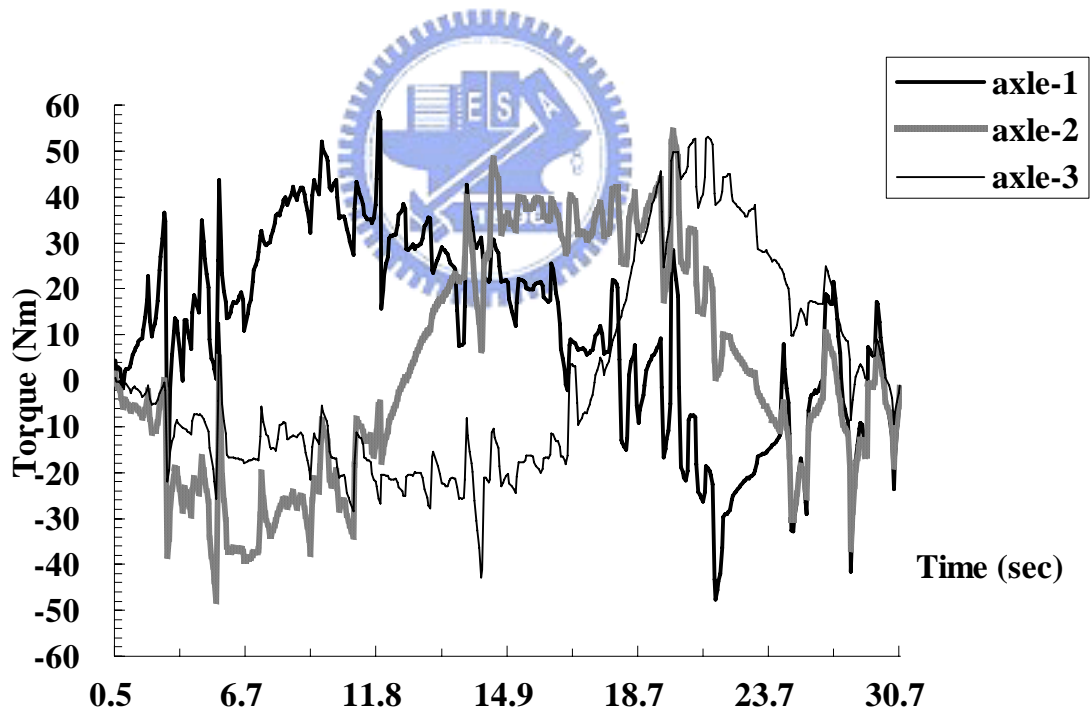


圖 5.17 力矩與舉重時間的關係圖(次方係數為 2)

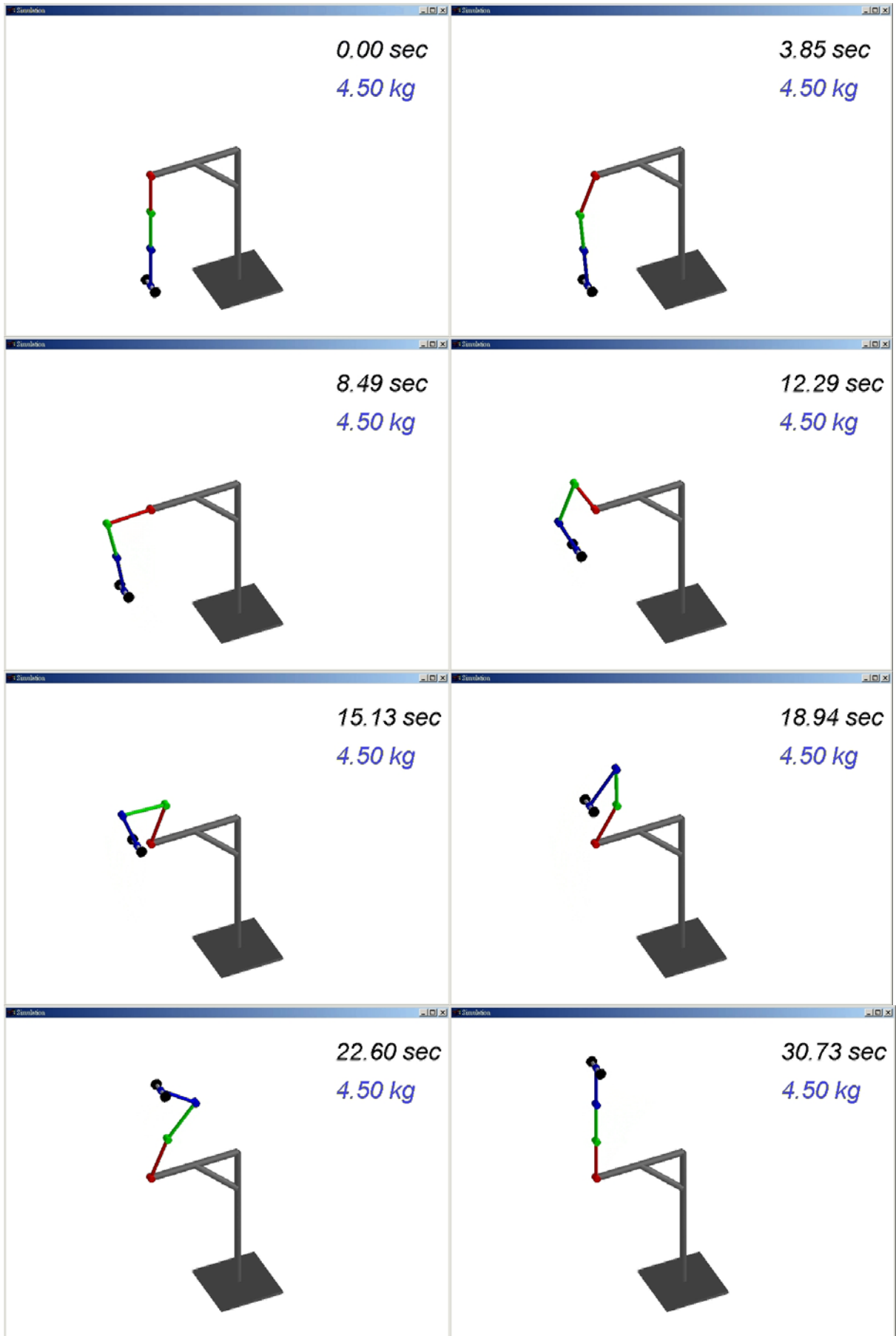


圖 5.18 動態模擬結果(次方係數為 2)



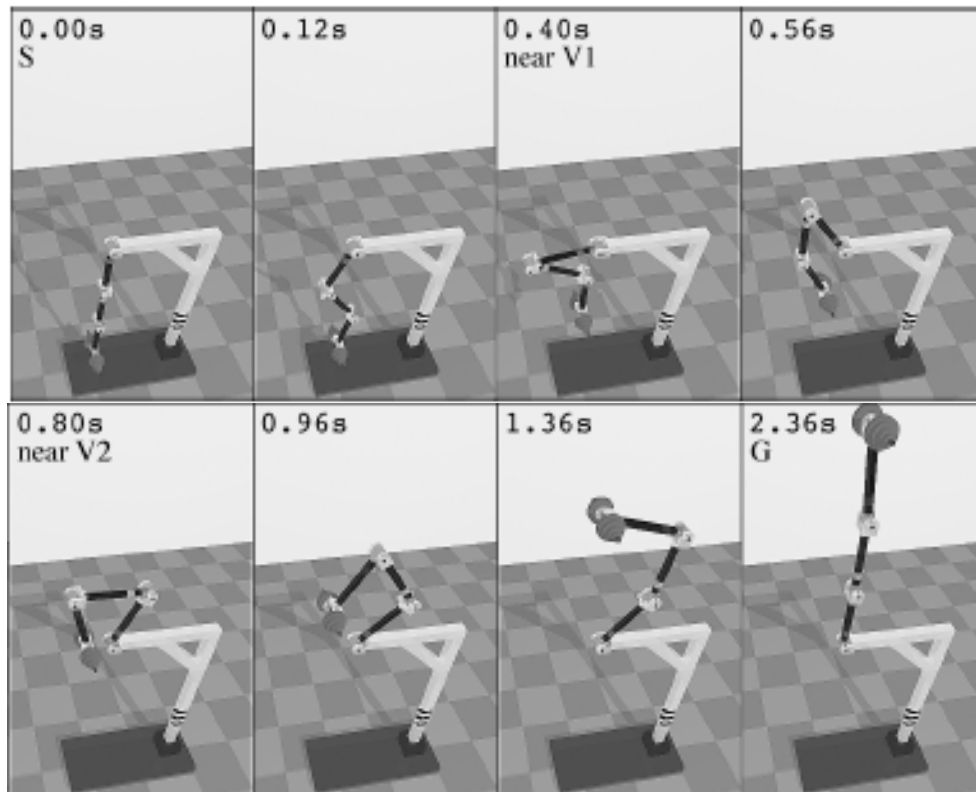


圖 5.19 文獻[4][5]的結果

由上面的結果來看，使用我們的方法所搜尋的路徑與文獻[4][5]的結果路徑很接近，不過我們舉重所花費的時間卻相當的緩慢。這是因為我們討論目標函數均為節點到節點求取最小值，再利用 Dijkstra 演算法來求取所有節點到節點的目標函數疊加最小的路徑。所以實際上我們在最小化節點間的目标函數值時並未考慮到整體性。這樣造成搜尋的結果中，節點到節點的經過時間都比較長，藉此得到較小的瞬間力矩值。

#### 5.4 結果討論

本章中主要研究以動力學來討論三軸機器手臂的路徑規劃問題，並結合 Dijkstra 演算法與 B-spline Curve 理論來搜尋具動力方程式的路徑。

首先根據 B-spline Curve 與 Curve fit 理論我們可以求得較精確的瞬間角速度、角加速度與力矩。接著根據積分梯形法則來求得節點到節點間的力矩對時間的積分最小值，最後利用第四章所提出的 Dijkstra 演算法加網格漸細法來求得目標函數疊加值最小的路徑。在 5.2 節中，我們使用上述的方法以得到初步良好的結果，使本論文所討論的三軸機器手臂能幾乎在限制條件內，來將重物由低處舉到高處。在 5.3 節中，我們修改機器手臂的尺寸與重量，使之與文獻[4][5]所討論的機器手臂相同，並以我們的方法搜尋結果來比較。雖然舉重花費的時間差異頗大，但是移動路徑卻十分類似，證明我們搜尋的路徑確實是一條比較省力的路徑。而且我們的方法不像文獻[4][5]需要參考中繼點的設定，也不像文獻[3]需要事先規劃初始路徑，所以相較之下我們的方法較為自動化。另外我們的方法搜尋的時間也比較快速，平均約一至兩個小時左右即可得到結果，也比文獻[3][4][5]來得較有效率。下面表 5.1 整理 5.2 與 5.3 節中，各個例子在搜尋結果所花費的時間。

表 5.1 本章中各個例子搜尋所花費的時間

搜尋例子	搜尋時間
5.2 節例子 1(次方係數為 1)	53 分 34 秒
5.2 節例子 2(次方係數為 2)	97 分 4 秒
5.3 節例子 1(次方係數為 2)	108 分 53 秒

雖然初步結果已經出來，不過仍然有需多地方需要我們去改進與修正。例如：(1)我們在做搜尋動作的時候，時間並不屬於搜尋空間中的一個維度，而是另外在決定節點的目標函數值的時候所計算決定的。所以即使是以網格漸細法作第二次以上搜尋的時候，節點間的經過時間計算都是由 0.001 秒到 10 秒作搜尋。這樣會造成網格尺寸越小則搜尋時間會過於冗長，而且最後結果的舉重時間會比較長。所以若能把時間參數加入搜尋空間，則我們就可以利用網格漸細法來減少搜尋的時間。(2)雖然用本論文的方法來作舉重機器手臂的路徑規劃，已經有不錯的結果。不過在總舉重時間與最大負重的表現上，尚且不如文獻[4][5]的成果。要改善這些缺點，除了更改目標函數的定義，另外也可以增加學習機制到 Dijkstra 演算法。使最後結果不理想或搜尋失敗時，可以學習並強化其表現，讓本文的路徑規劃方法更加完整、實用。