

機 器 人 學

機器人的軌跡規劃與防撞處理

內 容

- 簡介
- 軌跡規畫
- 軌跡控制法
- 追蹤控制法
- 路徑規畫

簡 介

機器人運動軌跡規劃的問題是指定一組位置點後，以指定的速度和加速度計畫通過有指定經過位置的一條軌跡。軌跡規劃要使軸關節轉動平滑，若轉動不平滑，可能產生不必要的衝擊運動，影響工作進行。

機器人的動作範圍是固定的，且在此範圍內可能有障礙物存在，爲了避免機器人碰撞造成損害，必須計算所規劃的軌跡是否可以安全通過障礙物。而在動作範圍的邊界，機器人是無法控制的，也要儘量避免將軸關節運動到這些點上。此外，爲了工作的需要也可能有一些路徑的限制，同樣影響到軌跡規劃的結果。

在軌跡規劃前要先進行路徑規劃，由於空間中的計算較複雜，所以先從移動式機器人在平面行走的問題開始討論。移動式機器人在平面上行走會遇到許多障礙物，從這些障礙物中找出一條能安全、快速通過的路徑，也是機器人防撞規劃的一個情況。再延伸下去，即爲智慧型機器人的問題。機器人不但要在已知環境中找到合適的路徑，更要能在未知或情報不完整的環境中利用本身的感測器辨識環境，找出合適的路徑。

軌跡規劃

- 使機器人運動的方法
 - 教導盒方式
 - 動作控制方式
- 軌跡規劃的方法簡介
 - 關節運動軌跡規劃
 - 直角運動軌跡規劃

軌跡規劃-使機器人運動的方法

➤ 教導盒方式

機器人先經引導，預先教導其作業內容，在移動的同時利用感測器將各軸的轉動資訊讀入並記憶之。若欲使機器人照原來的軌跡移動只需要回放原儲存之記憶內容即可動作。藉著“教導→記憶→重放”三步驟完成機器人動作的方法稱為教導盒方式或重放方式。有些機器人無法教導其動作速度，故重放動作時僅能以固定速度動作。

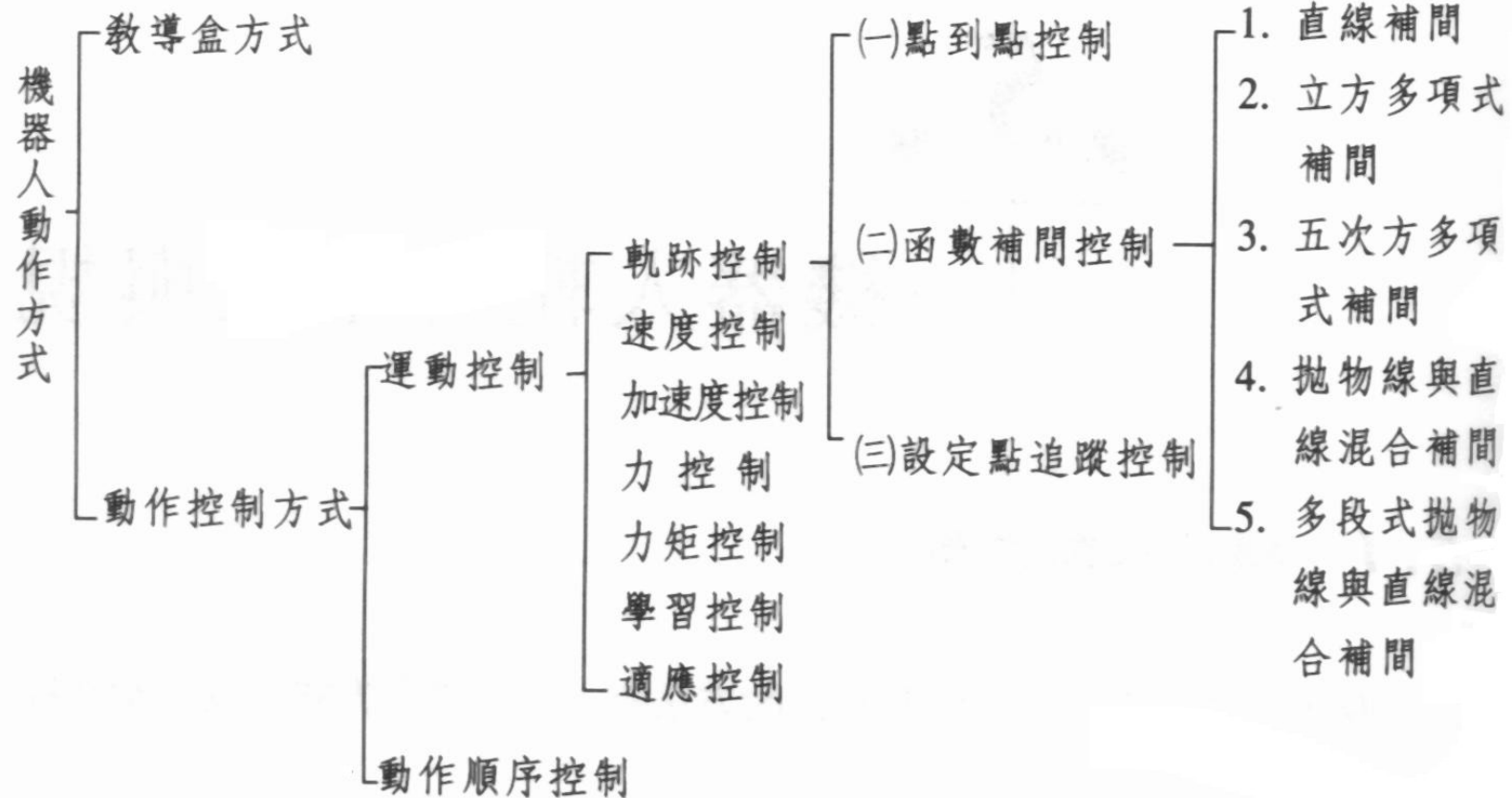
這種方法沒有一套完整的路徑規劃，使得機器人的操作無法完整實現工作空間軌跡規劃與直角座標的路徑控制，所以目前對機器人的軌跡規劃都以高階語言動作控制的方式為發展目標。

軌跡規劃-使機器人運動的方法

➤動作控制方式

動作控制可分為運動控制及動作順序控制。運動控制為控制機器人單一的動作，其中分為軌跡控制速度控制、加速度控制、力控制、力矩控制、學習控制及適應控制等。要控制機器人能夠完成所指定的工作，其中可能涉及許多不同的單一動作，規劃這些動作完成工作即為動作順序控制的探討內容。

軌跡規劃-機器人動作方式設計流程圖



軌跡規劃-軌跡和路徑的定義

機器人的軌跡是指機器人在運動期間其手部操作器在空間中所展現的位置、速度、加速度的情形，而軌跡規劃就是用來定義機器人手部操作器在空間中所行進的路徑與方位。

路徑是指手臂末端操作器從空間中一點移動至另一點的任意曲線。

軌跡是除了考慮二點之外，還需要考慮移動過程的位置、速度及加速度。所以路徑可有不同形態的軌跡，而軌跡只可表示一種路徑。

機器人軌跡產生，除了必須提供終點位置、速度及加速度外，通常還需要給予途經點等相關資料。機器人軌跡規劃是希望求得路徑中最佳的軌跡，即需要對其行經路徑進行路徑控制。

軌跡規劃-軌跡規劃的規則

- 運動軌跡正確通過指定的各位置設定點。
- 沿計劃軌跡運動時，機器人的速度、加速度連續，運動圓滑。

軌跡規劃-軌跡規劃的方法

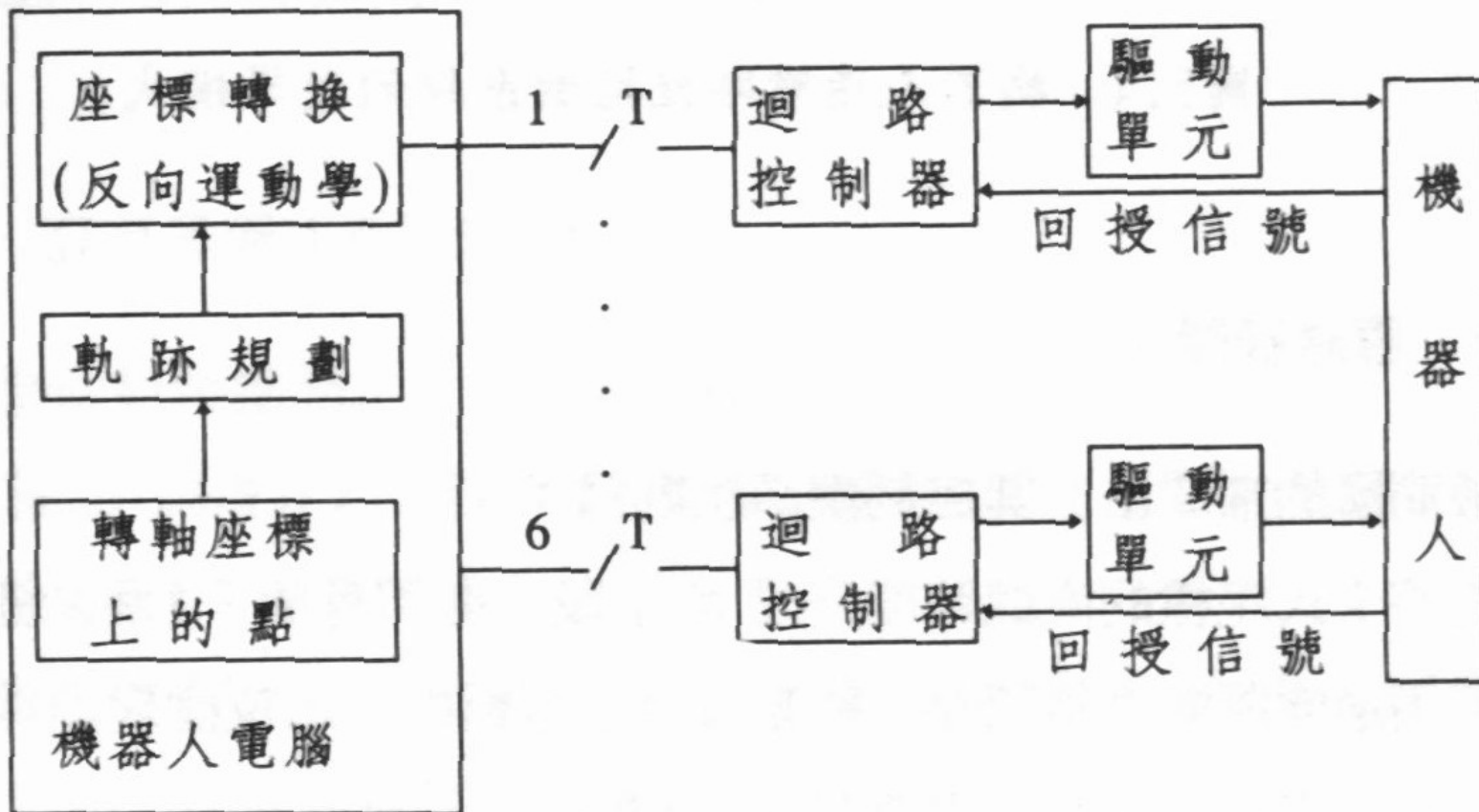
➤ 關節運動軌跡規劃

只要求機器人手部經過要求的設定點，至於設定點間的路徑即機器人手部的位置及姿態未加要求。其優點是計算簡單，缺點是運動期間各連桿位置不易判定。

➤ 直角運動軌跡規劃

它是以軸位置(轉角)來描述軌跡，所以無論起點、終點，各途經點都必須採用逆向運動學將卡式直角座標資料轉變成軸轉角資料方可進行軸座標控制，而每一軸隨著時間轉動資料均可以一平滑函數(如多項式)表示之，代表各軸運動的函數亦可不同。這種軌跡規劃方式的優點為(1)機器人手部平滑運動，(2)手部的位置與姿態在整個路徑上均有定義，(3)適用於工作空間有障礙物時，缺點為(1)要將空間路徑轉換成軸關節解，計算繁複，(2)計算中有奇異點產生。

軌跡規劃-路徑控制系統概念結構圖



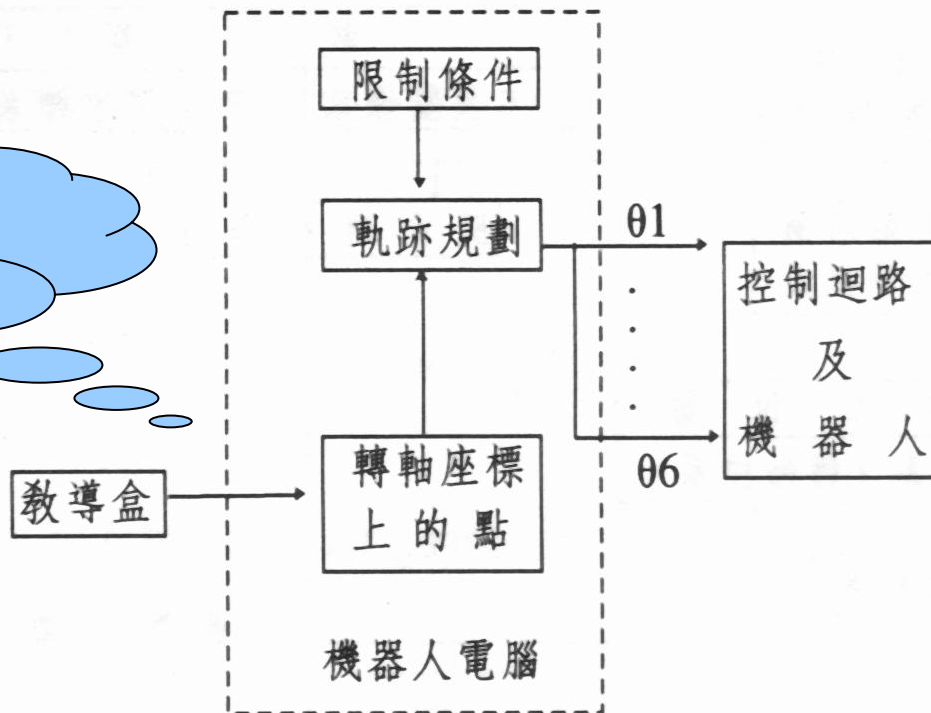
軌跡控制法

- 點到點控制法
- 函數補間控制法
- 追蹤控制法

軌跡控制法-點到點控制法

僅需提供下一個抵達點的位置給控制器，軌跡即可任意解得，這時應注意避免障礙物干擾，因該控制法無法對點到點路徑進行控制。通常可同時對多軸轉動以達成點到點間的移動。

關節型機器人點到點控制系統圖解



軌跡控制法-函數補間控制法

爲了避免手臂移動時超出工作空間容許範圍，須預先規畫路徑決定各途經點，然而少數的途經點不足以控制連續地軸關節運動，此時可使用函數補間法產生實際的軸關節命令。函數補間控制法與點到點控制法的最大差異爲可控制連續路徑。

常見的函數補間控制法有：

- (1)直線補間法
- (2)立方多項式補間法
- (3)五次方多項式補間法
- (4)拋物線與直線混合補間法
- (5)多段拋物線與直線混合補間法

軌跡控制法-函數補間控制法

直線補間法

控制演繹法則為：

- (1)先將卡式座標的直線路徑分為 K 小段，亦即有 $K-1$ 個途經點。
- (2)利用逆向運動學將起點、終點及各途經點的卡式位置變換為軸位置。
- (3)利用以下公式求出每一小段所需的時間

$$T_{seg} = \frac{\theta_{i_1} - \theta_{i_2}}{\omega_{i \max}} = \frac{\Delta\theta_i}{\omega_{i \max}}$$

其中 $\theta_{i_1}, \theta_{i_2}$ 為抵達每一小段兩端點所需的軸轉角， $\Delta\theta_i$ 為兩端點的轉角差，而 $\omega_{i \max}$ 代表軸 i 的最大角速度。
若為一定值，則軸 i 最大角速度會隨著中途點分佈距離的不同而改變。

軌跡控制法-函數補間控制法

直線補間法

(4)將段落時間 T_{seg} 分成 m 個等時距 dT_{seg} ， dT_{seg} 為取樣頻率 f_{sample} 的倒數，即

$$dT_{seg} = \frac{1}{f_{sample}} \quad m = \frac{T_{seg}}{dT_{seg}}$$

(5)決定每一軸在微小時距 dT_{seg} 的角位移 $d\theta_i$

$$d\theta_i = \frac{\Delta\theta_i}{m}$$

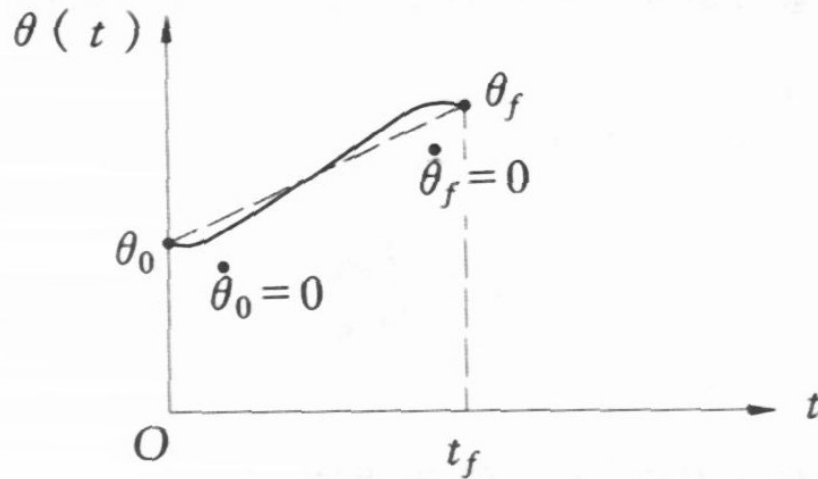
(6)在第 n 個取樣瞬間，伺服軸 i 收到的轉角指令應為 $n \cdot d\theta_i$

軌跡控制法 - 函數補間控制法

立方多項式補間法

(1) 連接兩個軌跡點間之立方多項式補間法

提供軸部4個邊界約束條件(如2個位置及2個速度)之軌跡，可以採用立方多項式含4個未知係數來表示之。



如左圖規劃一連接兩點的軸軌跡，已給起點及終點的軸位置分別為 θ_0 及 θ_f ，而速度均為0，即 $\dot{\theta}_0 = 0$ ， $\dot{\theta}_f = 0$ ，由此可建立第1種邊界約束條件為

$$BC'S: \begin{cases} \theta(0) = \theta_0 \\ \theta(t_f) = \theta_f \\ \dot{\theta}(0) = 0 \\ \dot{\theta}(t_f) = 0 \end{cases}$$

軌跡控制法-函數補間控制法

立方多項式補間法

以一立方多項式表示該軸軌跡，即

$$\theta(t) = a_0 + a_1t + a_2t^2 + a_3t^3$$

速度方程式為 $\dot{\theta}(t) = a_1 + 2a_2t + 3a_3t^2$

由邊界條件得

$$\begin{cases} a_0 = \theta_0 \\ a_3t_f^3 + a_2t_f^2 + a_1t_f + a_0 = \theta_f \\ a_1 = 0 \\ 3a_3t_f^2 + 2a_2t_f + a_1 = 0 \end{cases} \longrightarrow \begin{cases} a_0 = \theta_0 \\ a_1 = 0 \\ a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0) \\ a_3 = \frac{-2}{t_f^3}(\theta_f - \theta_0) \end{cases}$$

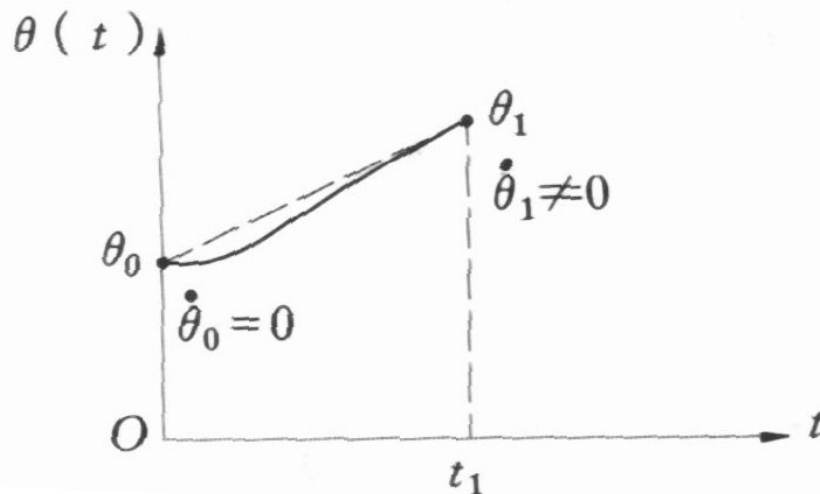
將上述解得係數代回原立方多項式的軸軌跡函數

$$\theta(t) = \frac{-2}{t_f^3}(\theta_f - \theta_0)t^3 + \frac{3}{t_f^2}(\theta_f - \theta_0)t^2 + \theta_0$$

軌跡控制法-函數補間控制法

立方多項式補間法

如右圖規劃一連接兩點的軸軌跡，已給起點及終點的軸位置分別為 θ_0 及 θ_f ，而速度為 $\dot{\theta}_0 = 0$ ， $\dot{\theta}_f \neq 0$ ，由此可建立第2種邊界約束條件為



$$BC'S : \begin{cases} \theta(0) = \theta_0 \\ \theta(t_1) = \theta_1 \\ \dot{\theta}(0) = 0 \\ \dot{\theta}(t_1) = \dot{\theta}_1 \end{cases}$$

第2種邊界約束條件的軸軌跡函數為

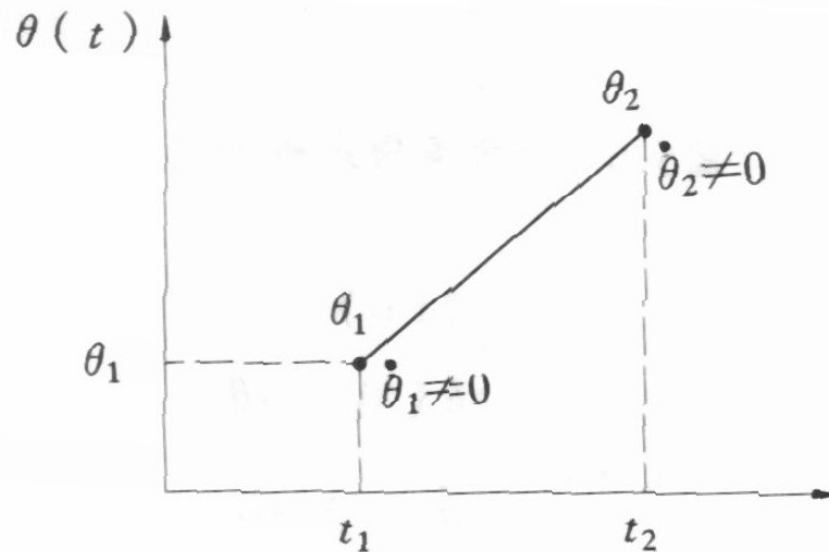
$$\theta(t) = \frac{-2(\theta_1 - \theta_0) + \dot{\theta}_1 t_1}{t_1^3} t^3 + \frac{-t_1 \dot{\theta}_1 + 3(\theta_1 - \theta_0)}{t_1^2} t^2 + \theta_0$$

軌跡控制法-函數補間控制法

立方多項式補間法

如右圖規劃一連接兩點的軸軌跡，已知軸的位置分別為 θ_1 ， θ_2 而速度為 $\dot{\theta}_1 \neq 0$ ， $\dot{\theta}_2 \neq 0$ ，由此可建立第3種邊界約束條件為

$$BC'S : \begin{cases} \theta(t_1) = \theta_1 \\ \theta(t_2) = \theta_2 \\ \dot{\theta}(t_1) = \dot{\theta}_1 \\ \dot{\theta}(t_2) = \dot{\theta}_2 \end{cases}$$



軌跡控制法-函數補間控制法

立方多項式補間法

第3種邊界約束條件的軸軌跡函數係數為

$$a_0 = \frac{(3t_1^2 t_2^2 - 4t_1 t_2^3 + t_2^4)\theta_1}{\Delta} + \frac{(t_1^4 - 4t_1^3 t_2 + 3t_1^2 t_2^2)\theta_2}{\Delta}$$
$$- \frac{(t_1^3 t_2^2 - 2t_1^2 t_2^3 + t_1 t_2^4)\dot{\theta}_1}{\Delta} - \frac{(t_1^4 t_2 - 2t_1^3 t_2^2 + t_1^2 t_2^3)\dot{\theta}_2}{\Delta}$$
$$a_1 = \frac{6t_1 t_2 (t_2 - t_1)(\theta_1 - \theta_2) + (2t_1^3 t_2 - 3t_1^2 t_2^2 + t_2^4)\dot{\theta}_1 + (t_1^4 - 3t_1^2 t_2^2 + 2t_1 t_2^3)\dot{\theta}_2}{\Delta}$$
$$a_2 = \frac{-3(t_2^2 - t_1^2)(\theta_1 - \theta_2) - (t_1^3 - 3t_1 t_2^2 + 2t_2^3)\dot{\theta}_1 - (2t_1^3 - 3t_1^2 t_2 + t_2^3)\dot{\theta}_2}{\Delta}$$
$$a_3 = \frac{2(t_2 - t_1)(\theta_1 - \theta_2) + (t_1 - t_2)^2(\dot{\theta}_1 + \dot{\theta}_2)}{\Delta}$$

其中 $\Delta = t_1^4 - 4t_1^3 t_2 + 6t_1^2 t_2^2 - 4t_1 t_2^3 + t_2^4 = (t_1 - t_2)^4$

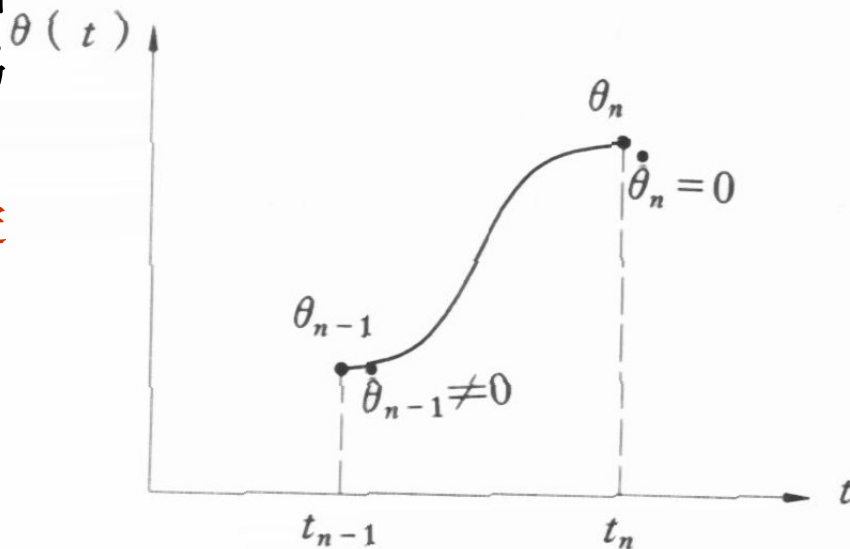
將上述解得係數代回原立方多項式，可得第3種邊界約束條件的軸軌跡函數。

軌跡控制法-函數補間控制法

立方多項式補間法

如右圖規劃一連接兩點的軸軌跡，已知軸的位置分別為 θ_{n-1} ， θ_n 而速度為 $\dot{\theta}_{n-1} \neq 0$ ， $\dot{\theta}_n = 0$ ，由此可建立第4種邊界約束條件為

$$BC'S : \begin{cases} \theta(t_{n-1}) = \theta_{n-1} \\ \theta(t_n) = \theta_n \\ \dot{\theta}(t_{n-1}) = \dot{\theta}_{n-1} \\ \dot{\theta}(t_n) = 0 \end{cases}$$



軌跡控制法-函數補間控制法

立方多項式補間法

第4種邊界約束條件的軸軌跡函數係數為

$$a_0 = \frac{(3t_{n-1}^2 t_n^2 - 4t_{n-1} t_n^3 + t_n^4)\theta_{n-1}}{\Delta} + \frac{(t_{n-1}^4 - 4t_{n-1}^3 t_n + 3t_{n-1}^2 t_n^2)\theta_n}{\Delta}$$
$$- \frac{(t_{n-1}^3 t_n^2 - 2t_{n-1}^2 t_n^3 + t_{n-1} t_n^4)\dot{\theta}_{n-1}}{\Delta}$$
$$a_1 = \frac{6t_{n-1} t_n (t_n - t_{n-1})(\theta_{n-1} - \theta_n) + (2t_{n-1}^3 t_n - 3t_{n-1}^2 t_n^2 + t_n^4)\dot{\theta}_{n-1}}{\Delta}$$
$$a_2 = \frac{-3(t_n^2 - t_{n-1}^2)(\theta_{n-1} - \theta_n) - (t_{n-1}^3 - 3t_{n-1} t_n^2 + 2t_n^3)\dot{\theta}_{n-1}}{\Delta}$$
$$a_3 = \frac{2(t_n - t_{n-1})(\theta_{n-1} - \theta_n) + (t_{n-1} - t_n)^2 \dot{\theta}_{n-1}}{\Delta}$$

其中 $\Delta = t_{n-1}^4 - 4t_{n-1}^3 t_n + 6t_{n-1}^2 t_n^2 - 4t_{n-1} t_n^3 + t_n^4 = (t_{n-1} - t_n)^4$

將上述解得係數代回原立方多項式，可得第4種邊界約束條件的軸軌跡函數。

軌跡控制法-函數補間控制法

立方多項式補間法

(2) 連接三個軌跡點間之立方多項式補間法

欲對包含途經點之立方多項式軸軌跡進行路徑控制，可將起點、途經點及終點的卡式座標位置利用逆向運動學變換為軸座標位置，而且各途經點之軸速度為已知值，則代表每一時段之立方多項式可根據上述不同之邊界條件推導求得軸解，最後再將代表各時段之立方多項式連接起來，即表示從起點出發經各途經點而抵達終點的軸軌跡。

軌跡控制法-函數補間控制法

立方多項式補間法

每一途經點的理想速度可採用下面3種方法求得：

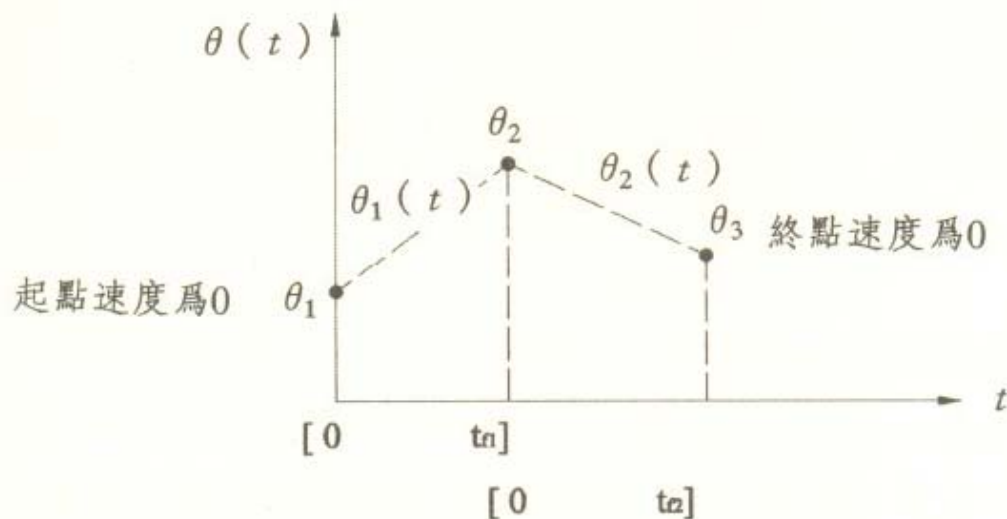
1. 指定每一點的卡式速度
2. 採簡單指定軸速度的法則求得途經點的軸速
3. 根據經過途經點之加速度為連續的觀念選取途經點速度

指定軸速法則為：

- (1) 當連接途經點兩直線斜率變號時，指定軸速度為0。
- (2) 當上述中斜率無變號，則取平均值作為該途經點的轉速。

軌跡控制法-函數補間控制法

立方多項式補間法



首先將連接3點的軸軌跡分為兩個段落，而每個段落之軸軌跡分別以一立方多項式表示之。

這裡分別以兩個立方多項式代表該兩個段落的軸軌跡，

$$\text{區間 } [0, t_{f1}] : \quad \theta_1(t) = a_{10} + a_{11}t + a_{12}t^2 + a_{13}t^3$$

$$\text{區間 } [0, t_{f2}] : \quad \theta_2(t) = a_{20} + a_{21}t + a_{22}t^2 + a_{23}t^3$$

軌跡控制法-函數補間控制法

立方多項式補間法

已知條件爲：

- (1) 3個軸軌跡點爲 $\theta_1, \theta_2, \theta_3$
- (2) 起點及終點速度爲0
- (3) 途經點 θ_2 的速度與加速度爲連續

仿照前述兩軌跡點的方式，建立8個邊界約束條件。8個邊界條件可用兩個立方多項式解之，即軸軌跡函數 $\theta_1(t), \theta_2(t)$ 解

$$BC'S : \begin{cases} \theta_1(0) = \theta_1, & \dot{\theta}_1(0) = 0 \\ \theta_1(t_{f1}) = \theta_2, & \dot{\theta}_1(t_{f1}) = \dot{\theta}_2(0) \\ \theta_2(0) = \theta_2, & \dot{\theta}_2(0) = 0 \\ \theta_2(t_{f2}) = \theta_3, & \ddot{\theta}_2(t_{f2}) = \ddot{\theta}_2(0) \end{cases}$$

軌跡控制法-函數補間控制法

五次方多項式補間法

當某段軌跡已給起點及終點的位置、速度及加速度時，即具**6**個邊界約束條件，因此可以一個五次方多項式做為該段軌跡的函數解。故假設軸軌跡為

$$\theta(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

其速度及加速度分別為

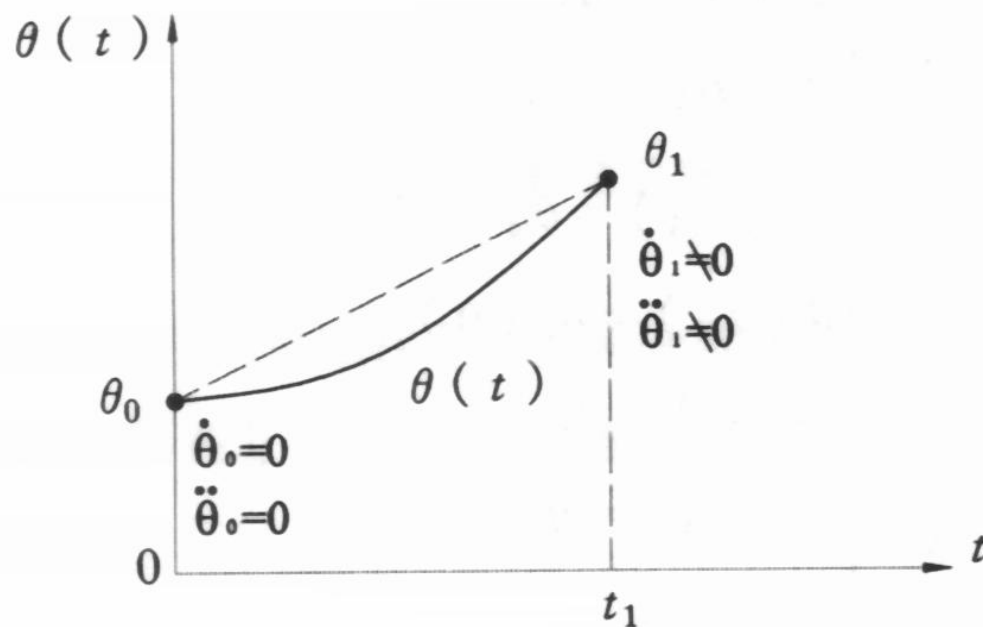
$$\dot{\theta}(t) = 5a_5 t^4 + 4a_4 t^3 + 3a_3 t^2 + 2a_2 t + a_1$$

$$\ddot{\theta}(t) = 20a_5 t^3 + 12a_4 t^2 + 6a_3 t + 2a_2$$

軌跡控制法-函數補間控制法

五次方多項式補間法

(1) 連接兩點的軌跡邊界條件為：起點速度及加速度為0，終點速度加速度均不為0。



$$BC'S : \begin{cases} \theta(0) = \theta_0 \\ \theta(t_1) = \theta_1 \\ \dot{\theta}(0) = 0 \\ \dot{\theta}(t_1) = \dot{\theta}_1 \\ \ddot{\theta}(0) = 0 \\ \ddot{\theta}(t_1) = \ddot{\theta}_1 \end{cases}$$

軌跡控制法-函數補間控制法

五次方多項式補間法

由邊界條件得

$$\begin{cases} a_0 = \theta_0 \\ a_5 t_1^5 + a_4 t_1^4 + a_3 t_1^3 + a_2 t_1^2 + a_1 t_1 + a_0 = \theta_1 \\ a_1 = 0 \\ 5a_5 t_1^4 + 4a_4 t_1^3 + 3a_3 t_1^2 + 2a_2 t_1 + a_1 = \dot{\theta}_1 \\ a_2 = 0 \\ 20a_5 t_1^3 + 12a_4 t_1^2 + 6a_3 t_1 + 2a_2 = \ddot{\theta}_1 \end{cases}$$

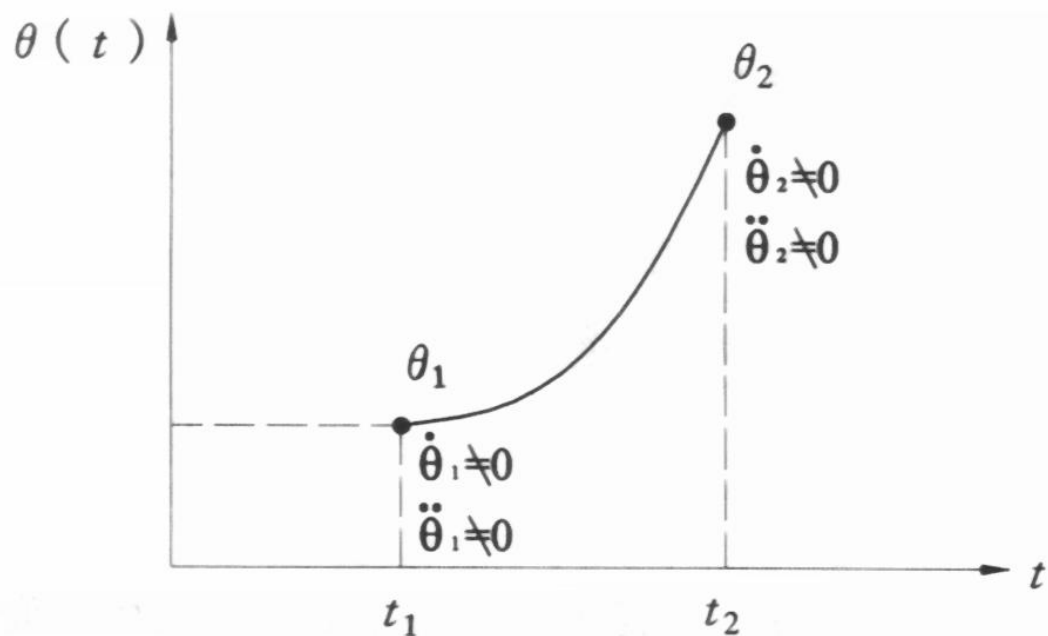
解聯立方程式後，可得軌跡方程式的係數為

$$\begin{cases} a_5 = \frac{6}{t_1^5}(\theta_1 - \theta_0) - \frac{3}{t_1^4}\dot{\theta}_1 + \frac{1}{2t_1^3}\ddot{\theta}_1 \\ a_4 = -\frac{15}{t_1^4}(\theta_1 - \theta_0) + \frac{7}{t_1^3}\dot{\theta}_1 - \frac{1}{t_1^2}\ddot{\theta}_1 \\ a_3 = \frac{10}{t_1^3}(\theta_1 - \theta_0) - \frac{4}{t_1^2}\dot{\theta}_1 - \frac{1}{2t_1}\ddot{\theta}_1 \\ a_2 = 0 \\ a_1 = 0 \\ a_0 = \theta_0 \end{cases}$$

軌跡控制法-函數補間控制法

五次方多項式補間法

(2) 連接兩點的軌跡邊界條件為：起點與終點速度加速度均不為0。



$$BC'S : \begin{cases} \theta(t_1) = \theta_1 \\ \theta(t_2) = \theta_2 \\ \dot{\theta}(t_1) = \dot{\theta}_1 \\ \dot{\theta}(t_2) = \dot{\theta}_2 \\ \ddot{\theta}(t_1) = \ddot{\theta}_1 \\ \ddot{\theta}(t_2) = \ddot{\theta}_2 \end{cases}$$

軌跡控制法-函數補間控制法

五次方多項式補間法

由邊界條件得

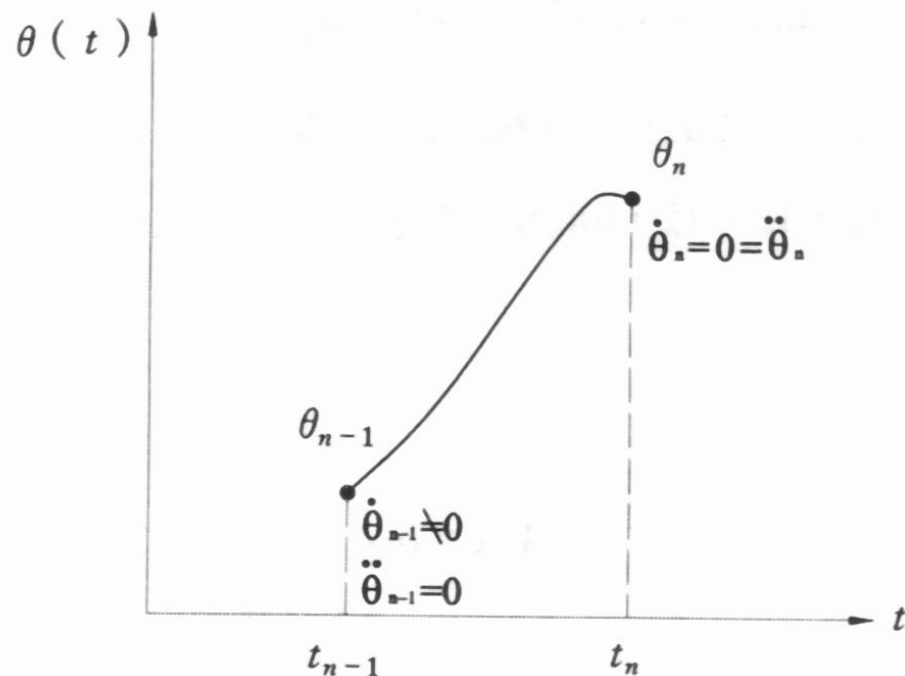
$$\begin{cases} a_5 t_1^5 + a_4 t_1^4 + a_3 t_1^3 + a_2 t_1^2 + a_1 t_1 + a_0 = \theta_1 \\ a_5 t_2^5 + a_4 t_2^4 + a_3 t_2^3 + a_2 t_2^2 + a_1 t_2 + a_0 = \theta_2 \\ 5a_5 t_1^4 + 4a_4 t_1^3 + 3a_3 t_1^2 + 2a_2 t_1 + a_1 = \dot{\theta}_1 \\ 5a_5 t_2^4 + 4a_4 t_2^3 + 3a_3 t_2^2 + 2a_2 t_2 + a_1 = \dot{\theta}_2 \\ 20a_5 t_1^3 + 12a_4 t_1^2 + 6a_3 t_1 + 2a_2 = \ddot{\theta}_1 \\ 20a_5 t_2^3 + 12a_4 t_2^2 + 6a_3 t_2 + 2a_2 = \ddot{\theta}_2 \end{cases}$$

解聯立方程式後，可得軌跡方程式的係數。

軌跡控制法-函數補間控制法

五次方多項式補間法

(3) 連接兩點的軌跡邊界條件為：終點速度及加速度為0，起點的速度不為0加速度為0。



$$BC'S : \begin{cases} \theta(t_{n-1}) = \theta_{n-1} \\ \theta(t_n) = \theta_n \\ \dot{\theta}(t_{n-1}) = \dot{\theta}_{n-1} \\ \dot{\theta}(t_n) = 0 \\ \ddot{\theta}(t_{n-1}) = \ddot{\theta}_{n-1} \\ \ddot{\theta}(t_n) = 0 \end{cases}$$

軌跡控制法-函數補間控制法

五次方多項式補間法

由邊界條件得

$$\begin{cases} a_5 t_{n-1}^5 + a_4 t_{n-1}^4 + a_3 t_{n-1}^3 + a_2 t_{n-1}^2 + a_1 t_{n-1} + a_0 = \theta_{n-1} \\ a_5 t_n^5 + a_4 t_n^4 + a_3 t_n^3 + a_2 t_n^2 + a_1 t_n + a_0 = \theta_n \\ 5a_5 t_{n-1}^4 + 4a_4 t_{n-1}^3 + 3a_3 t_{n-1}^2 + 2a_2 t_{n-1} + a_1 = \dot{\theta}_{n-1} \\ 5a_5 t_n^4 + 4a_4 t_n^3 + 3a_3 t_n^2 + 2a_2 t_n + a_1 = 0 \\ 20a_5 t_{n-1}^3 + 12a_4 t_{n-1}^2 + 6a_3 t_{n-1} + 2a_2 = \ddot{\theta}_{n-1} \\ 20a_5 t_n^3 + 12a_4 t_n^2 + 6a_3 t_n + 2a_2 = 0 \end{cases}$$

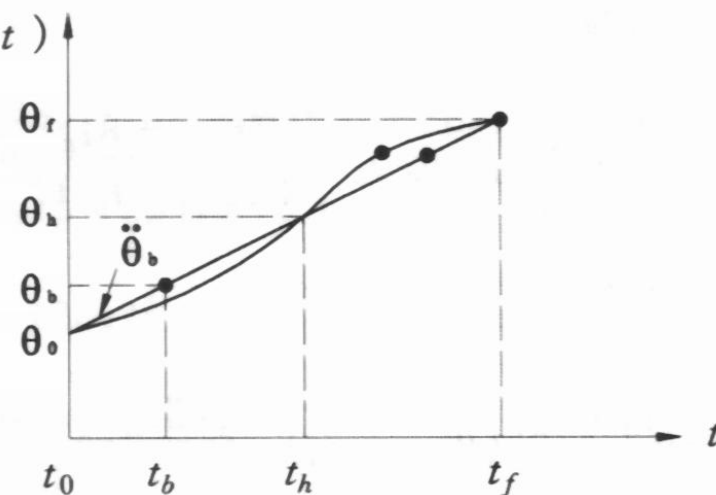
解聯立方程式後，可得軌跡方程式的係數。

軌跡控制法-函數補間控制法

拋物線與直線混合補間法

爲使軌跡起點與終點的位置及速度連續，採兩拋物線平滑地與一直線相銜接，此謂拋物線與直線混合補間法。

已知軸起始轉角 θ_0 ，最終轉角 θ_f 及整個軌跡所需時間 t ($t = t_f - t_0$)，現在先選擇拋物線段想要的軸加速度 $\ddot{\theta}_b$ ，接著由拋物線與直線的連接公式與拋物線方程式求得第一個拋物線結束所需的時間 t_b 。



軌跡控制法-函數補間控制法

拋物線與直線混合補間法

第一個拋物線結束時速度必須等於連接於直線的速度，即

$$v_{tb} = \dot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b} \Rightarrow \theta_h - \theta_b = \dot{\theta}_b t_b (t_h - t_b)$$

$$\Rightarrow 2(\theta_h - \theta_b) = 2\dot{\theta}_b t_b (t_h - t_b) \text{————— (a)}$$

第一個拋物線方程式建立為 $\theta(t) = \theta_0 + \frac{1}{2}\ddot{\theta}_b t^2$

$$(1) t = t_0 \Rightarrow \dot{\theta}_b(t_0) = 0, \theta(t_0) = \theta_0$$

$$(2) t = t_b \Rightarrow \theta(t_b) = \theta_b = \theta_0 + \frac{1}{2}\ddot{\theta}_b t_b^2 \Rightarrow 2(\theta_b - \theta_0) = \ddot{\theta}_b t_b^2 \text{————— (b)}$$

$$(a)+(b) \Rightarrow 2(\theta_h - \theta_0) = 2\dot{\theta}_b t_b t_h - \ddot{\theta}_b t_b^2 \text{————— (c)}$$

$$\text{假設 } t_f = 2t_h \Rightarrow \theta_f - \theta_0 = 2(\theta_h - \theta_0) \text{————— (d)}$$

軌跡控制法-函數補間控制法

拋物線與直線混合補間法

$$(d) \text{ 代入(c)} \Rightarrow \ddot{\theta}_b t_b^2 - 2\ddot{\theta}_b t_h t_b + (\theta_f - \theta_0) = 0$$

令軸轉角時間為 t ，且 $t = 2t_h$ ，則第一個拋物線結束所需的時間為

$$t_b = \frac{t}{2} - \frac{\sqrt{\ddot{\theta}_b^2 - 4\theta_b(\theta_f - \theta_0)}}{2\ddot{\theta}_b}$$

第一個拋物線結束所需的時間若要有解，必須滿足以下條件

$$\ddot{\theta}_b \geq \frac{4(\theta_f - \theta_0)}{t^2}$$

討論：

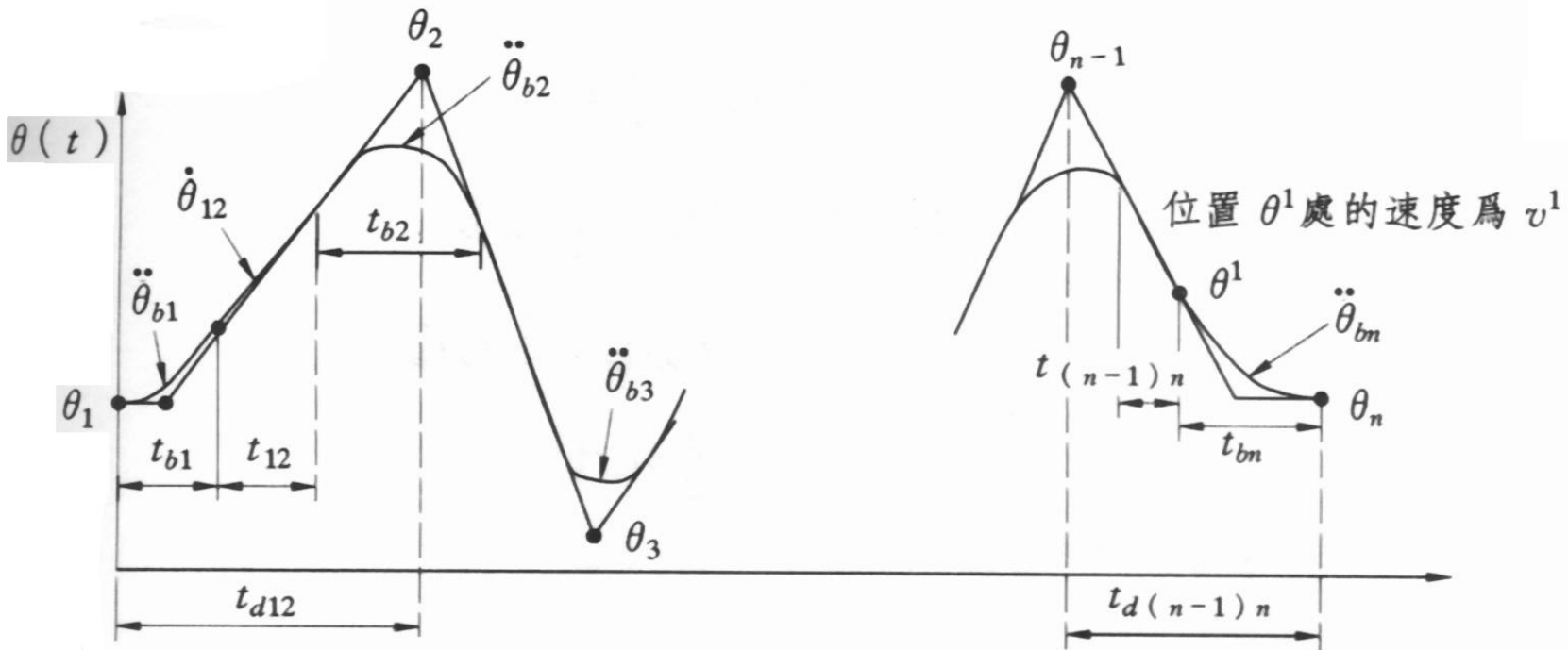
$$(1) \quad \ddot{\theta}_b = \frac{4(\theta_f - \theta_0)}{t^2} \Rightarrow \text{兩條拋物線}$$

$$(2) \quad \ddot{\theta}_b > \frac{4(\theta_f - \theta_0)}{t^2} \Rightarrow \text{拋物線與直線}$$

$$(3) \quad \ddot{\theta}_b \gg \frac{4(\theta_f - \theta_0)}{t^2} \Rightarrow \text{直線}$$

軌跡控制法-函數補間控制法

多段拋物線與直線混合補間法



軌跡控制法-函數補間控制法

多段拋物線與直線混合補間法

(1)起點 θ_1 拋物線域所需時間 t_{b1} 的求法，由速度恆等式得

$$\ddot{\theta}_{b1} t_{b1} = \frac{\theta_2 - \theta_1}{t_{d12} - \frac{1}{2} t_{b1}}$$

$$\Rightarrow \ddot{\theta}_{b1} t_{b1} t_{d12} - \frac{1}{2} \ddot{\theta}_{b1} t_{b1}^2 = \theta_2 - \theta_1$$

$$\Rightarrow \ddot{\theta}_{b1} t_{b1}^2 - 2\ddot{\theta}_{b1} t_{b1}^2 + 2(\theta_2 - \theta_1) = 0$$

$$\Rightarrow t_{b1} = t_{d12} - \sqrt{t_{d12}^2 - \frac{2(\theta_2 - \theta_1)}{\ddot{\theta}_{b1}}}$$

軌跡控制法-函數補間控制法

多段拋物線與直線混合補間法

(2) 終點 θ_n 拋物線域所需時間 t_{bn} 的求法，由速度恆等式得

$$v' = \frac{\Delta\theta}{\Delta t} \quad \Rightarrow \quad \ddot{\theta}_{bn} t_{bn} = \frac{\theta_{n-1} - \theta_n}{t_{d(n-1)n} - \frac{1}{2} t_{bn}}$$

$$\Rightarrow \ddot{\theta}_{bn} t_{bn} t_{d(n-1)n} - \frac{1}{2} \ddot{\theta}_{bn} t_{bn}^2 = \theta_{n-1} - \theta_n$$

$$\Rightarrow \ddot{\theta}_{bn} t_{bn}^2 - 2\ddot{\theta}_{bn} t_{d(n-1)n} t_{bn} - 2(\theta_n - \theta_{n-1}) = 0$$

$$\Rightarrow t_{bn} = t_{d(n-1)n} - \sqrt{t_{d(n-1)n}^2 + \frac{2(\theta_n - \theta_{n-1})}{\ddot{\theta}_{bn}}}$$

軌跡控制法-函數補間控制法

多段拋物線與直線混合補間法

(3) 轉角位置 θ_1 與 θ_2 間直線段所需時間 t_{12} 為

$$t_{12} = t_{d12} - t_{b1} - \frac{1}{2}t_{b2}$$

(4) θ_1 與 θ_2 直線域的速度 $\dot{\theta}_{12}$ 為

$$\dot{\theta}_{12} = \frac{\theta_2 - \theta_1}{t_{d12} - \frac{t_{b1}}{2}}$$

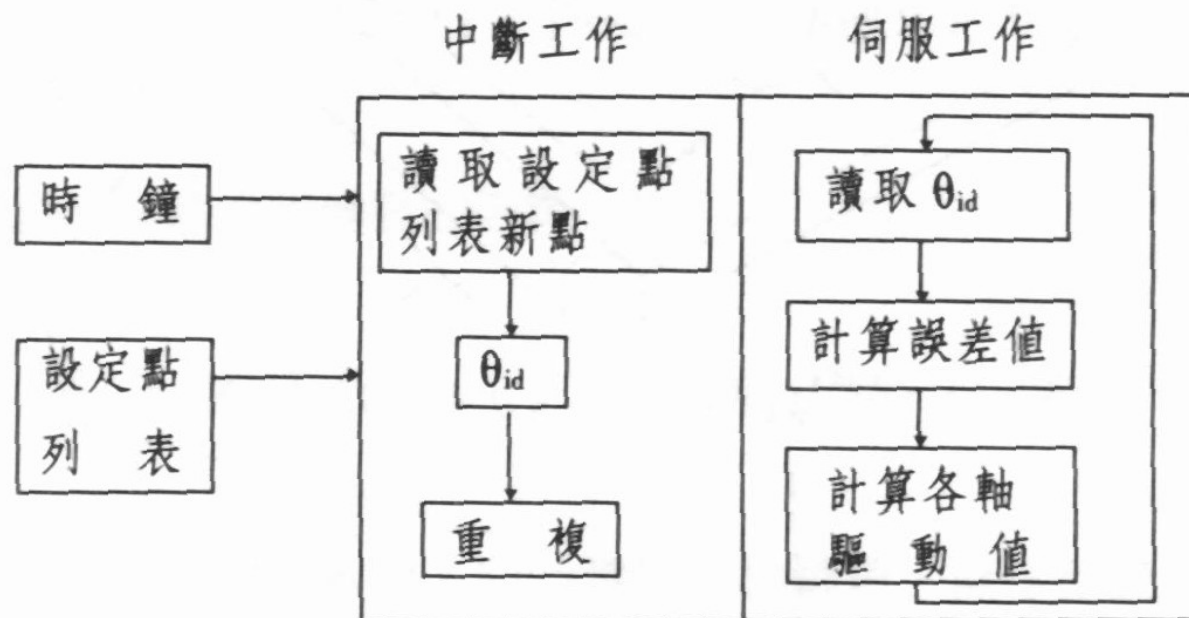
(5) 在轉角位置 θ_2 拋物線域所需時間 t_{b2} 的求法

$$t_{b2} = \frac{\dot{\theta}_{23} - \dot{\theta}_{12}}{\ddot{\theta}_2}$$

追蹤控制法-設定點對點追蹤控制

(1)線上即時追蹤法：

可同步進行中斷及伺服工作。即當關節軸運轉時，同時計算各軸所需之驅動函數值及各軸理想之位置。



追蹤控制法-設定點對點追蹤控制

(2)線外計算路徑追蹤法：

在移動前於線外進行軌跡規劃，定出軌跡表(設定各點間時距)。利用逆向運動學轉換將卡式空間位置轉換為關節軸理想應至之位置 θ_{id} ，並輸入至設定點列表中儲存，當重放時即可依列表順序進行工作。該追蹤法又稱為批次追蹤法，即預先將教導資料一次計算解決。

追蹤控制法-卡式追蹤控制

當機器人移動或工件移動時行進的各路徑點必須即時轉換為軸座標。而卡式控制可允許工件位置為時間函數(如輸送帶上工件)及 / 或外在感測器為時間函數(如電視攝影機)。這使得卡式控制特別適用於輸送帶上工件追蹤的問題。

路徑規劃

路徑規劃問題是在已知的環境中，找出從出發點到目標點的路線，有時要求該路線滿足某種性質。數學上的定義是：路徑規劃是指在線圖構造中，找出滿足規定性質的路徑。

線圖(graph)構造是指有關環境的各種情報中，只抽出與路徑規劃有關者，表現成線圖。因而有時已知有關線圖構造的詳細情報，有時幾無有關線圖構造的情報。具備有關線圖構造的知識時，或可藉之增高路徑探索效率。毫無知識時，只有以試行錯誤的方式持續探索，用所得知識與累積的知識，增高探索的效率。

路徑規劃

一般而言，解決路徑規劃問題有3種方法，一種是**拓撲法**、一種是**幾何法**、另一種是**勢場法**。這些概念來自於人類解決複雜問題規劃的思路。

機器人與環境的原始關係是障礙物分佈在空間中，而機器人本身有一定的尺寸，也要占據一定的空間。這種關係處理起來較為困難。所以一般討論路徑規劃時，首先要對原始關係進行轉換。通常使用的方法是“障礙膨脹—機器人縮小”的方法。若機器人為圓形，則將障礙物往外膨脹圓型半徑大小，而機器人縮小為一個質點。若機器人為一長方體，則將障礙物往外膨脹機器人寬度的一半大小，而機器人縮小為一杆狀物，即寬度為0，長度不變的等效機器人。

路徑規劃

人類在進行路徑選擇時，不是首先考慮障礙物分布的精確位置，而是將環境劃分為拓撲特徵一致的若干子區域，根據彼此的連通性建立一個網路。首先在網路上尋找一條拓撲路徑，然後在進一步考慮幾何細節，確定具體的幾何路徑。在機器人行走的過程中，對於意外發生的障礙，或是由於誤差所引起的環境變化，一般還要用勢場法進行局部防撞規劃，以補償幾何規劃中求出路徑的誤差。

路徑規劃 - 拓樸法

拓樸法可分為下列4個步驟：

(1) 區域分割

考慮到機器人在空間中的轉動，將空間分為3種類型。一種是自由空間，機器人在空間中可以自由轉動而不發生碰撞。一種是半自由空間，機器人只能在有限的角度轉動。一種是障礙空間，機器人不允許進入。在障礙較多的情況下，空間的劃分算法十分複雜，用解析算法很難甚至不可能實現。一般只能通過人機交互，再利用計算圖形學計算出若干特性參數。例如能自由轉動角度的集合，稱為方向分支結構，從而得出區域分割的結果。

(2) 建立拓樸網路

區域分割，每個子區域對應於旋轉映射圖的一種方向分支結構。逐個追蹤每個子空間的邊界搜索及相鄰的每個子空間，計算彼此的連通性(即其方向分支的交集非空集合)，形成連通網路。

路徑規劃 - 拓樸法

(3) 搜索拓樸路徑

在拓樸網路上搜索拓樸路徑。可以利用人工智能中的各種演算法，對於不太複雜的網路結構，也可以採用寬度優先、深度優先等簡單算法。

(4) 求出幾何路徑

對於拓樸路徑已經確定，每段拓樸路徑只涉及兩個區域或同一區域，環境相對簡單可直接用幾何法求解。

路徑規劃 - 幾何法

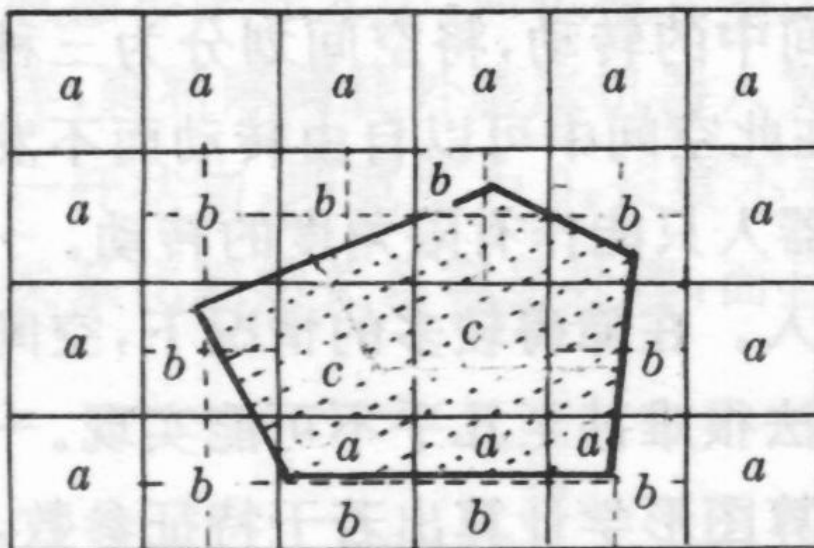
以下介紹幾何法常用的**3**種方法：

- (1)網格法
- (2)廣義錐法
- (3)輪廓空間法

路徑規劃 - 幾何法

網格法

網格法是先將空間分爲若干粗網格，由於有障礙存在，網格可以標示爲3種情況之一，有自由網格，半自由網格，障礙網格。



左圖中，*a*代表自由網格，*b*代表半自由網格，*c*代表障礙網格。

路徑規劃 - 幾何法

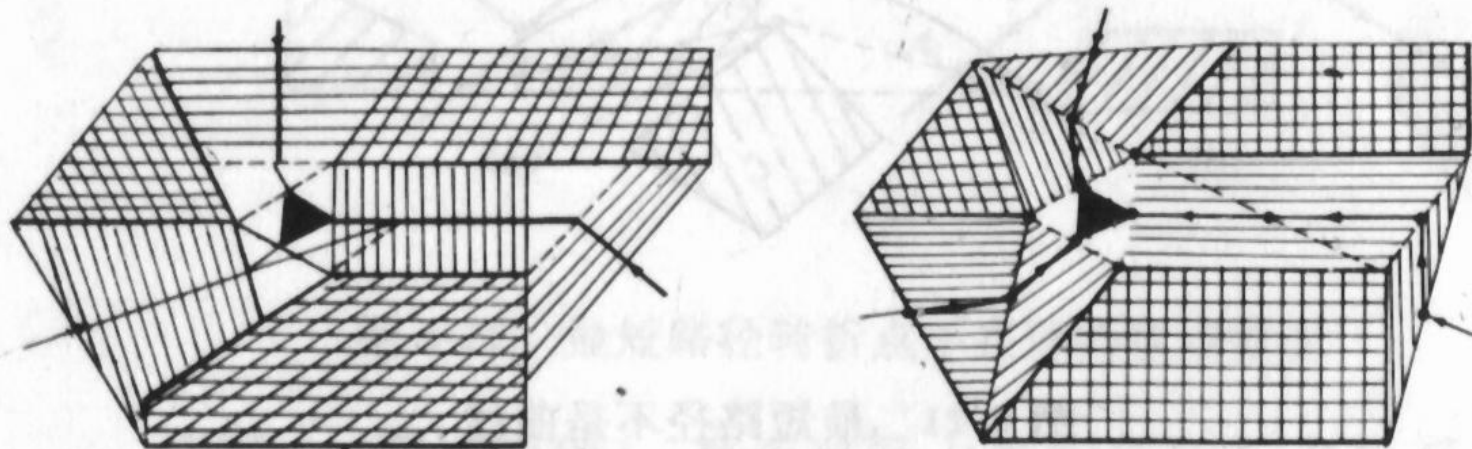
網格法

對半自由網格還可進一步細分，例如每網格再細分為4個子網格，這些子網格再標識出3種類型(a、b、c)。對b類型再繼續細分為4個子網格，又可標識出三種類型，如此反覆繼續下去，直到一定的精度為止。對於退化為質點的機器人而言，規劃問題就是在這些大小不等的a類連通網格之間尋找起點到終點的通路。對於退化成桿狀的機器人而言，連通性除了要保證平面上的連接之外，還必須保證網格中，桿狀機器人自由轉動角度的連續性。這個問題又變成前面討論的拓撲特性一致子空間的連通性問題。問題的複雜性由二維平面問題變成三維空間問題(第三坐標是轉動角度)。

路徑規劃 - 幾何法

廣義錐法

廣義錐法的空間思想是將自由空間分成若干個錐形區。如下圖所示。機器人行走路徑在錐形區的中心線上。這樣可以保證最大的安全性，避免與障礙物發生碰撞。



路徑規劃 - 幾何法

廣義錐法

尋找有意義錐形區的算法很複雜，特別是在計算機上自動進行時，很難判斷哪兩條邊將組成有意義的錐形區，若採用人機交互的方法也許是解決問題的途徑，但只能離線進行，限制了它的應用性。

廣義錐法的變種是三角形法。它將自由空間分成若干個相互鄰接的三角形區，機器人的安全行走路徑是在這些三角形的腰線上。

透過廣義錐法或三角形法，求出最安全路徑的集合，考察起始點和目標點落在哪個錐形區，用搜索方法就能容易求出最安全路徑。

路徑規劃 - 幾何法

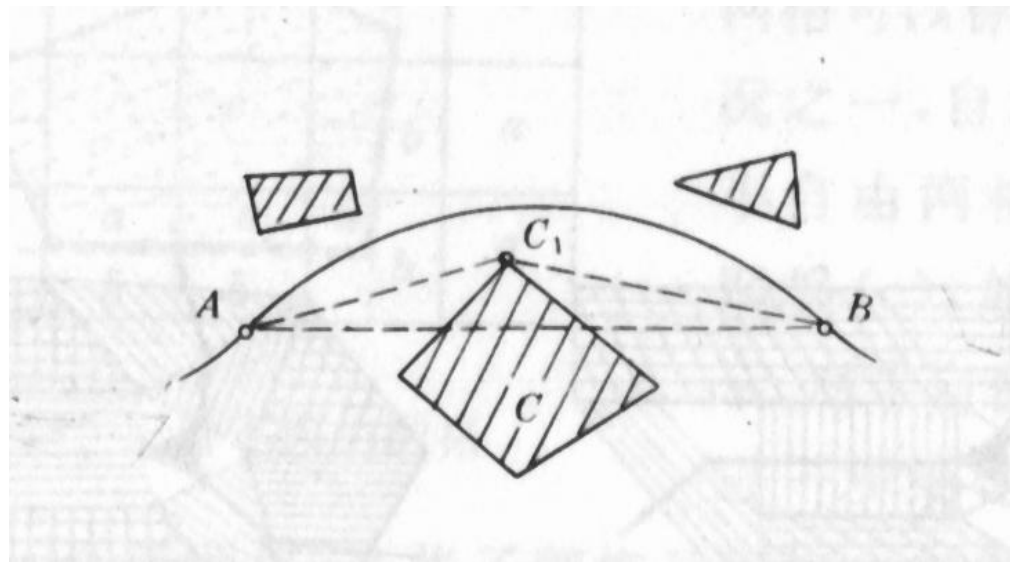
輪廓空間法

輪廓空間法的基本原理是：從起點到終點的最短路徑必然是由通過某些障礙物的頂點的折線所組成。

常用的原理：

(1) 折線原理

最短路徑由自由空間內的折線組成，而不包括任何曲線段。

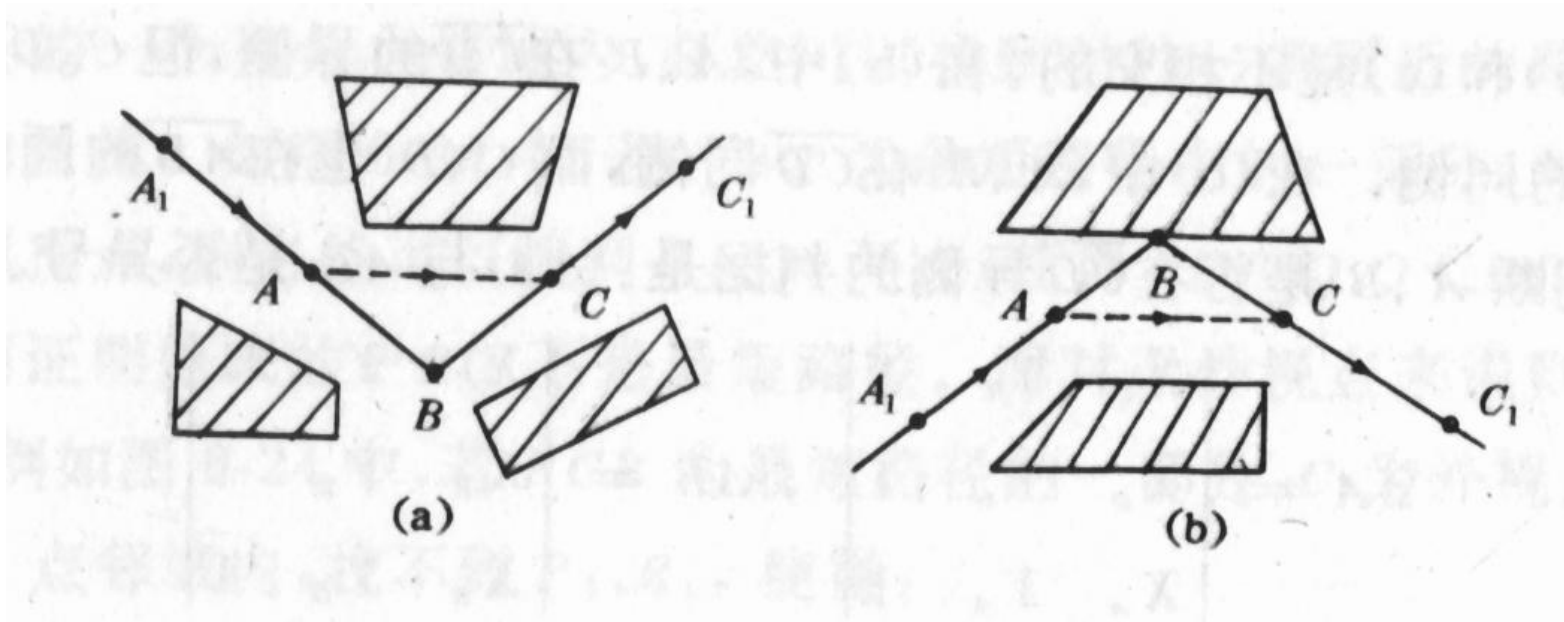


路徑規劃 - 幾何法

輪廓空間法

(2) 頂點原理

最短路徑的轉折點必然位於障礙物的頂點處。

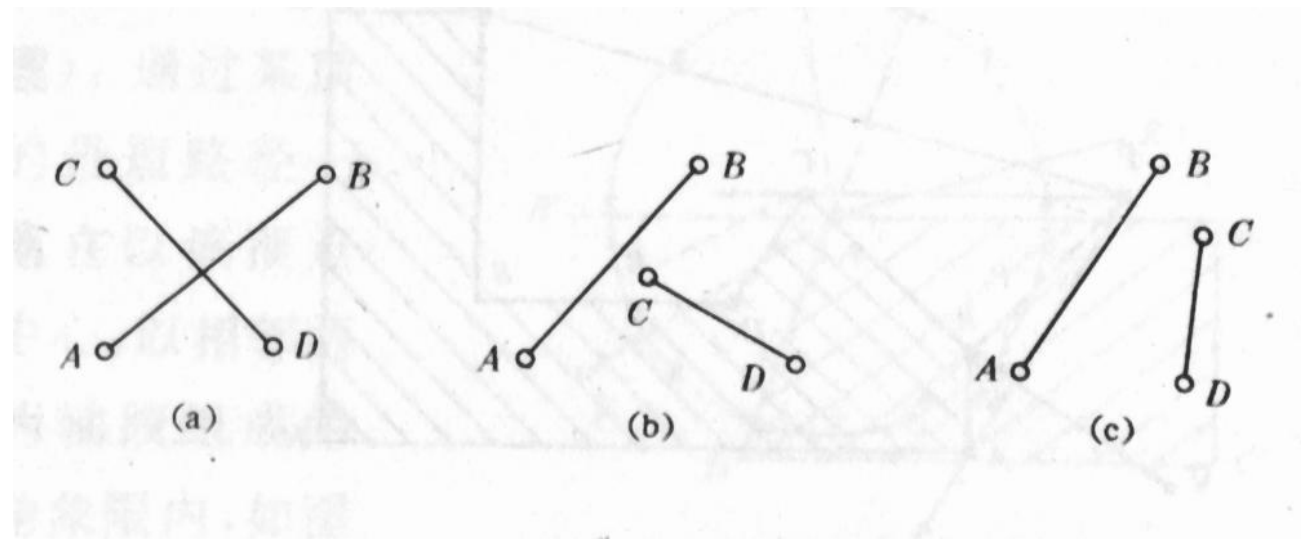


路徑規劃 - 幾何法

輪廓空間法

(3) 可視原理

最短路徑上兩相鄰轉折點是互相可視的。判斷兩點之間是否可視，最簡單的方法是檢查兩點連線段是否與任何一條障礙的邊相交。若相交，表示兩點連線穿過障礙物，因此是不可視的，否則就是可視的。

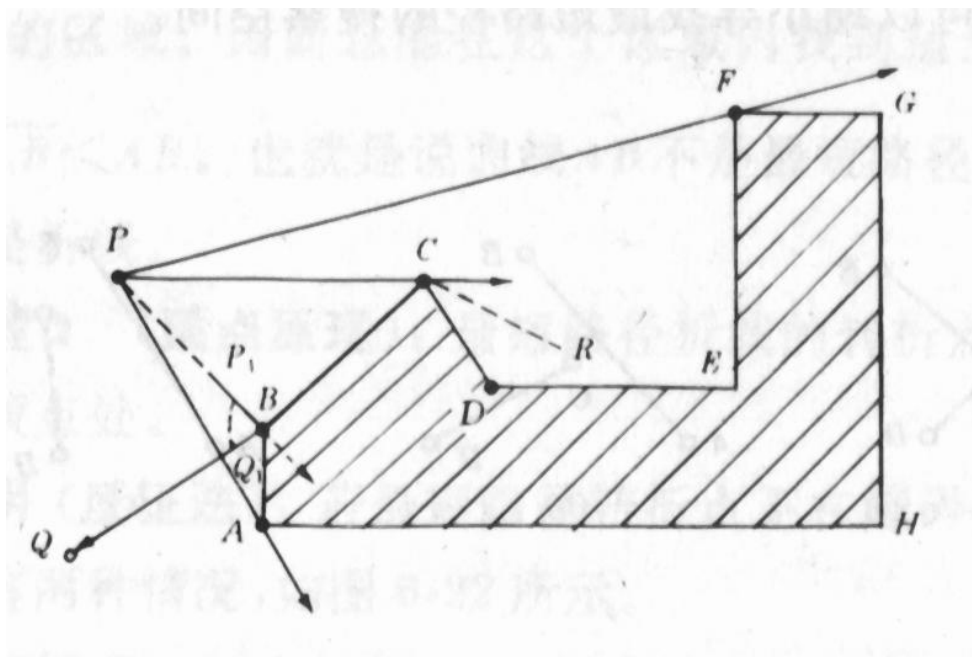


路徑規劃 - 幾何法

輪廓空間法

(4) 外視原理

最短路徑是聯結外視頂點的折線所組成。所謂外視頂點是指當從某點觀察障礙頂點時，其視線延長線若不進入障礙物內部，而只從頂點掠過的那些點。



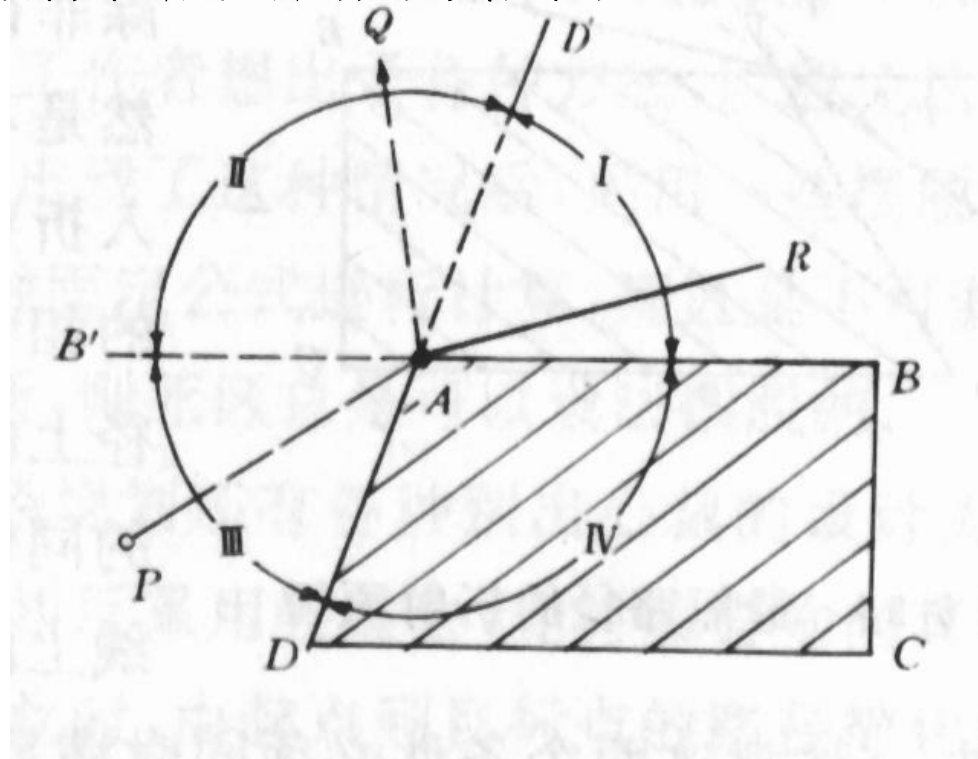
判斷內視、外視的準則是：障礙上的相鄰頂點是否在視線的同側還是異側。若同側，則為外視點，若異側則為內視點。

路徑規劃 - 幾何法

輪廓空間法

(5) 對角原理

通過某頂點的最短路徑一定落在以該頂點為中心，以相鄰兩邊為軸線組成的對角象限內。



路徑規劃 - 幾何法

輪廓空間法

(6) 折射原理

穿入對角象限內的最短路徑，必然折向障礙物一邊。

