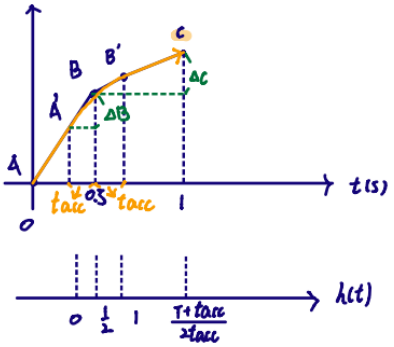
機器人學_Project2

311512019 江柏霖

1. 介面說明
   - 開發平台: matlab
   - Joint move: Joint_move.m ，點擊 run 鍵即可產生結果
   - Cartesian move: Cartesian_move.m ，點擊 run 鍵即可產生結果

2. 程式架構說明
   - Joint_move:
     先將 A,B,C 三點帶入 inverse kinemics 獲得各個點對應的 joint variables，再將這些 joint variables 進行直線與 transition 的路徑規劃，透過 forward kinemics 得到卡氏座標之路徑，最後輸出結果
   - Cartesian move:
     先將 A,B,C 三個 transformation matrix 轉換成卡氏座標的 x, y ,z, ϕ, θ, ψ，將這些 variables 進行直線與 transition 的路徑規劃，最後輸出結果

3. 數學運算說明



$$A=\begin{bmatrix}0&0&-1&10\\-1&0&0&20\\0&1&0&30\\0&0&0&1\end{bmatrix}\quad B=\begin{bmatrix}1&0&0&30\\0&1&0&20\\0&0&1&10\\0&0&0&1\end{bmatrix}$$

$$C=\begin{bmatrix}0&-1&0&-10\\0&0&1&-20\\-1&0&0&-30\\0&0&0&1\end{bmatrix}\quad (cm)$$

$t_{acc}=0.25$, $T=1$

sampling time $=0.0025$

1' linear portion (A~A' , B'~C)

set $h=\frac{t}{T}$

then $q(t)=\Delta C\,h+B$ .. position

$\dot{q}(t)=\frac{\Delta C}{T}$ .. velocity

$\ddot{q}(t)=0$ .. acceleration

2° transition portion (A'~B')

set $t_B = 0$, $h(t) = \frac{t + t_{acc}}{2t_{acc}}$, $-t_{acc} < t < t_{acc}$

$$\begin{cases} q(h) = a_4h^4 + a_3h^3 + a_2h^2 + a_1h + a_0 \\ \dot{q}(h) = 4a_4h^3 + 3a_3h^2 + 2a_2h + a_1 \\ \ddot{q}(h) = 12a_4h^2 + 6a_3h + 2a_2 \end{cases}$$

$$\begin{cases} q(0) = a_0 = A \longrightarrow a_0 = B + \Delta B \\ \dot{q}(0) = a_1 = -\Delta B / \frac{1}{2} = -2\Delta B \longrightarrow a_1 = -2\Delta B \\ \ddot{q}(0) = 2a_2 = 0 \longrightarrow a_2 = 0 \end{cases} \quad \begin{bmatrix} \Delta B = A' - B \\ \Delta C = C - B \end{bmatrix}$$

.. boundary condition

$$\begin{cases} \dot{q}(1) = 4a_4 + 3a_3 + 2a_2 + a_1 = \frac{\Delta C}{\frac{T + t_{acc}}{2t_{acc}} - \frac{1}{2}} = \frac{2\Delta C \cdot t_{acc}}{T} \\ \ddot{q}(1) = 12a_4 + 6a_3 + 2a_2 = 0 \end{cases}$$

.. boundary condition

$$\longrightarrow a_3 = 2\left(\Delta C \cdot \frac{t_{acc}}{T} + \Delta B\right)$$

$$\longrightarrow a_4 = -\left(\Delta C \cdot \frac{t_{acc}}{T} + \Delta B\right)$$

position : $q(t) = q(h(t)) = \left[\left(\Delta C \cdot \frac{t_{acc}}{T} + \Delta B\right)(2 - h)h^2 - 2\Delta B\right]h + B + \Delta B$

velocity : $\dot{q}(t) = \dot{q}(h(t)) \cdot \frac{dh}{dt} = \left[\left(\Delta C \cdot \frac{t_{acc}}{T} + \Delta B\right)(1.5 - h) \cdot 2h^2 - \Delta B\right] \cdot \frac{1}{t_{acc}}$

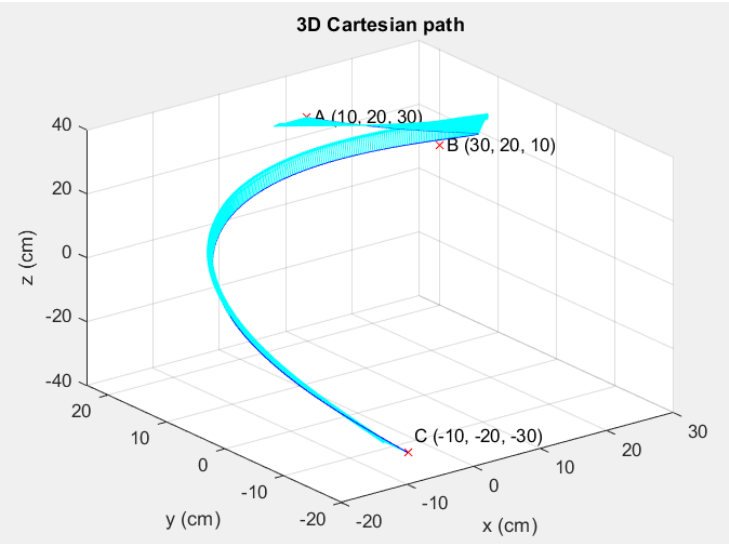
acceleration : $\ddot{q}(t) = \ddot{q}(h(t))\left(\frac{dh}{dt}\right)^2 + \dot{q}(t)\frac{dh^2}{dt^2}$

$$= \left[\left(\Delta C \cdot \frac{t_{acc}}{T} + \Delta B\right)(1 - h)\right] \cdot \frac{3h}{t_{acc}}$$

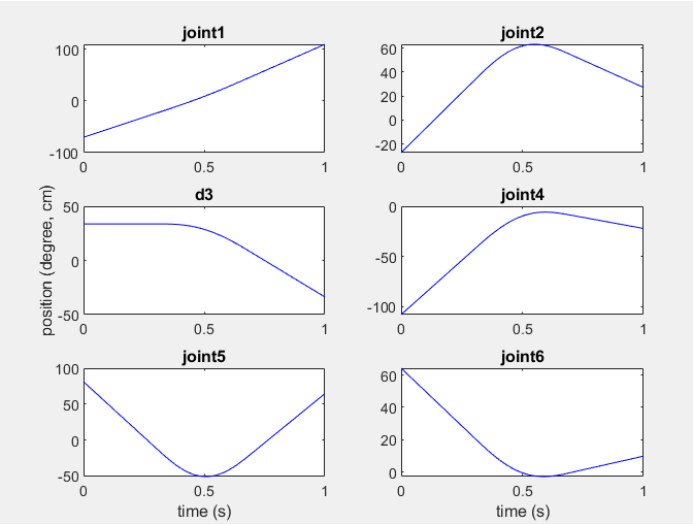
4. 軌跡規劃結果

- Joint_move:

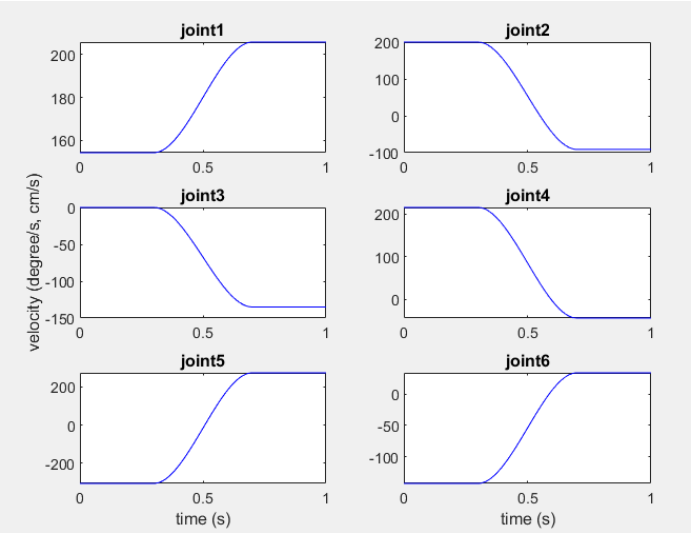
  3D Cartesian path



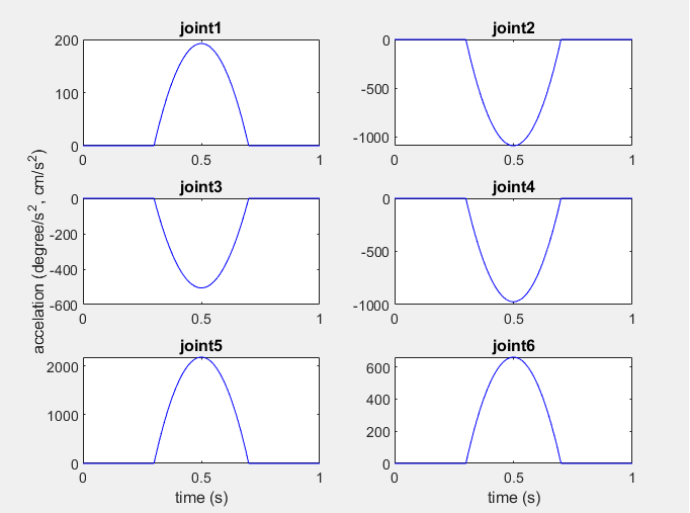
Joint position



Joint velocity

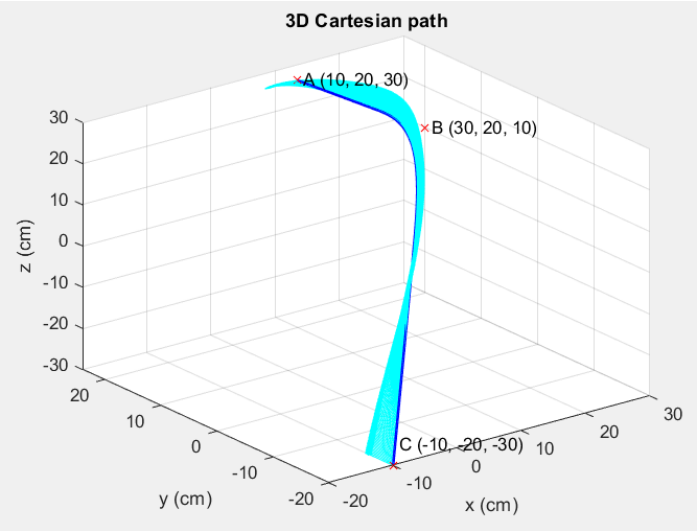


Joint acceleration

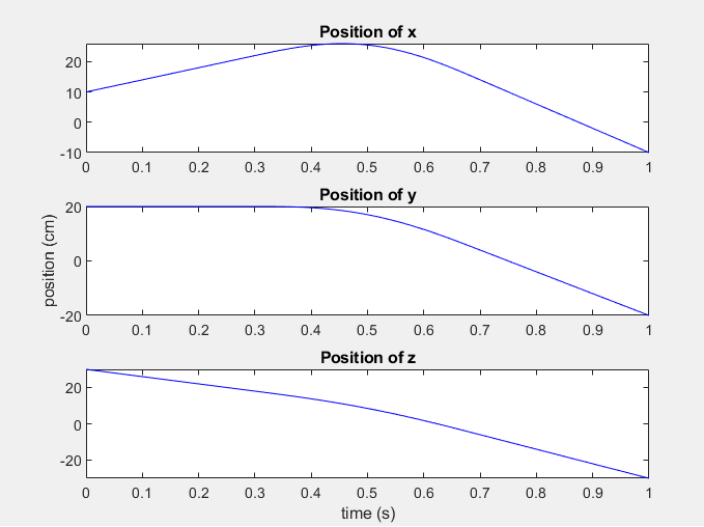


- Cartesian move:

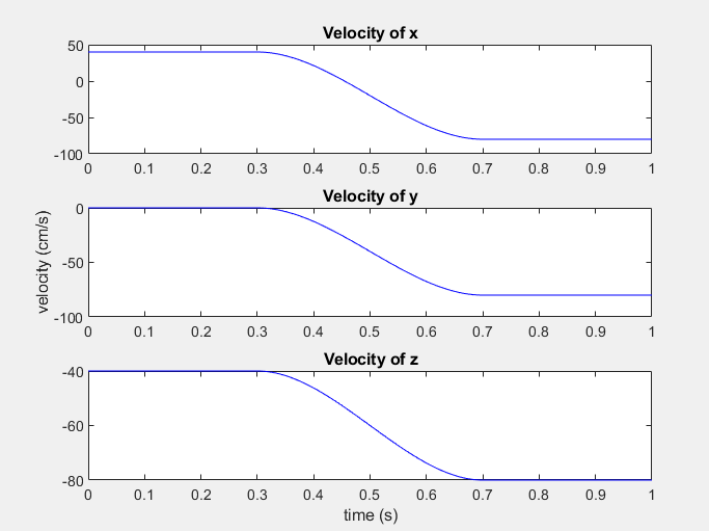
  3D Cartesian path



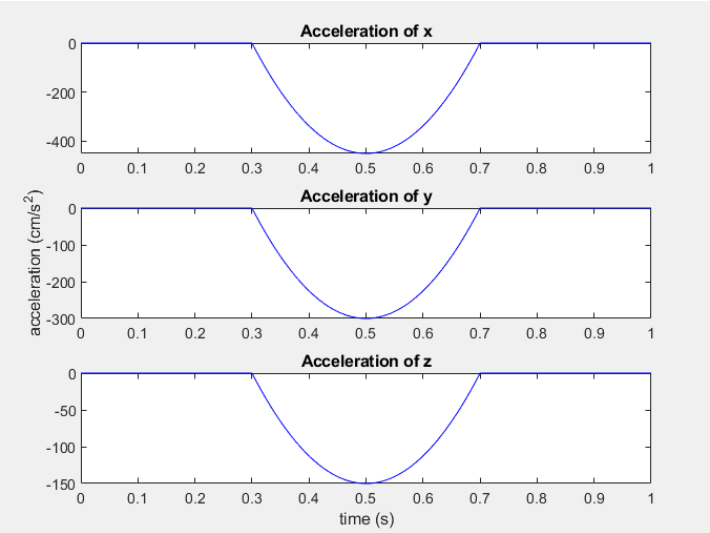
Cartesian position



Cartesian velocity



Cartesian acceleration



5. 優缺點
   - Joint_move:
     優點: 不用 inverse 所以計算效率高，不會有碰到 singular point 的問題
     缺點: 卡氏座標的路徑可能很複雜
   - Cartesian move:
     優點: 路徑之間的定義較為明確
     缺點: 較高的計算量，可能碰上 singular point