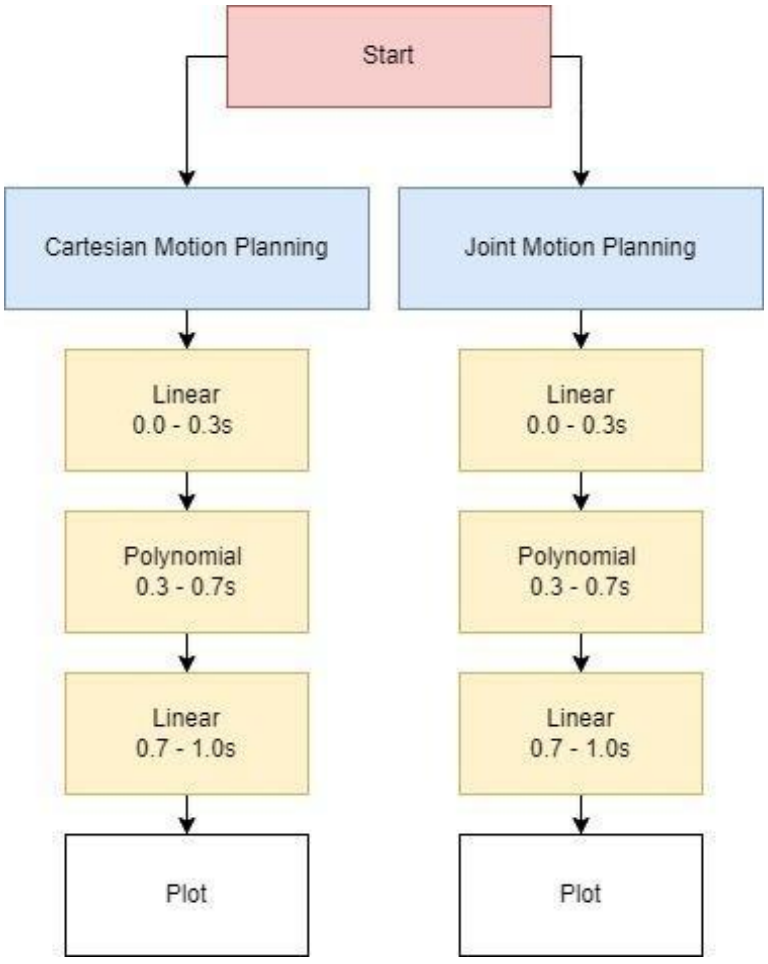
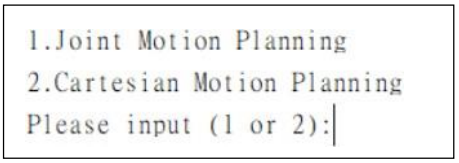
機器人學 Project 2

310605015 機器人碩一 李啟安

I. 程式架構（使用 Matlab）

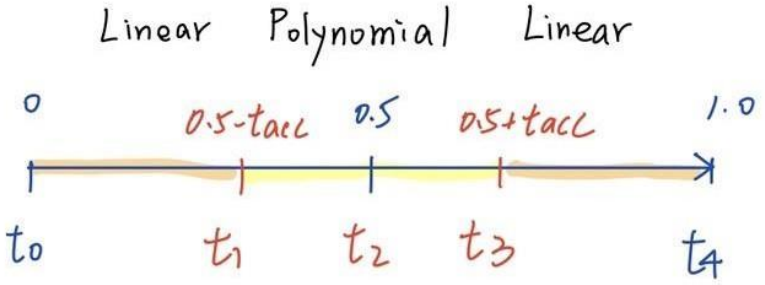
1. 一鍵執行
2. 可以看到會詢問使用者要使用 Joint motion planning 或是 Cartesian motion planning
3. 選擇 1 或是 2 ，就會開始進行路徑規劃





II. 數學運算

Joint Move



A. (前後2段 Linear 段，使用等速度移動)

$\Delta C = C - B \quad \Delta B = B - A$

① $t_0 \sim t_1$

$h = t / t_2$

6個角度

$angle = \Delta B \cdot h + A$

$angle\_vel = \Delta B / t_2$

$angle\_acc = 0$

② $t_3 \sim t_4$

$h = \frac{t - t_2}{t_4 - t_2}$

$angle = \Delta C \cdot h + B$

$angle\_vel = \Delta C / (t_4 - t_2)$

$angle\_acc = 0$

B. $t_1 \sim t_3$ 做 2次加速度規劃

③ $t_1 \sim t_3$

$h = \frac{t - t_1}{t_3 - t_1}$

$angle: \left(\left(\Delta C \cdot \frac{tacc}{t_2} - \Delta B \cdot \frac{tacc}{t_2}\right)(2-h)h^2 + 2\Delta B \cdot \frac{tacc}{t_2}\right)h + B - \Delta B \frac{tacc}{t_2}$

$angle\_vel: \left(\left(\Delta C \cdot \frac{tacc}{t_2} - \Delta B \cdot \frac{tacc}{t_2}\right) \cdot (1.5-h) \cdot 2h^2 + \Delta B \frac{tacc}{t_2}\right) \cdot \frac{1}{tacc}$

$angle\_acc: \left(\Delta C \frac{tacc}{t_2} - \Delta B \cdot \frac{tacc}{t_2}\right) \cdot (1-h) \cdot \frac{3h}{tacc^2}$

Cartesian Move

方法和 Joint move 一致. input 從 六軸 angle → X.Y.Z.Ψ.Θ.Φ

A. (前後2段 Linear 段, 使用等速度移動)

$\Delta C = C - B \quad \Delta B = B - A$

① $t_0 \sim t_1$

$h = t / t_2$

$angle = \Delta B \cdot h + A$ (X,Y,Z,Ψ,Θ,Φ)

$angle\_vel = \Delta B / t_2$

$angle\_acc = 0$

② $t_3 \sim t_4$

$h = \frac{t - t_2}{t_4 - t_2}$

$angle = \Delta C \cdot h + B$

$angle\_vel = \Delta C / (t_4 - t_2)$

$angle\_acc = 0$

B. $t_1 \sim t_3$ 做 2次加速度規劃

③ $t_1 \sim t_3$

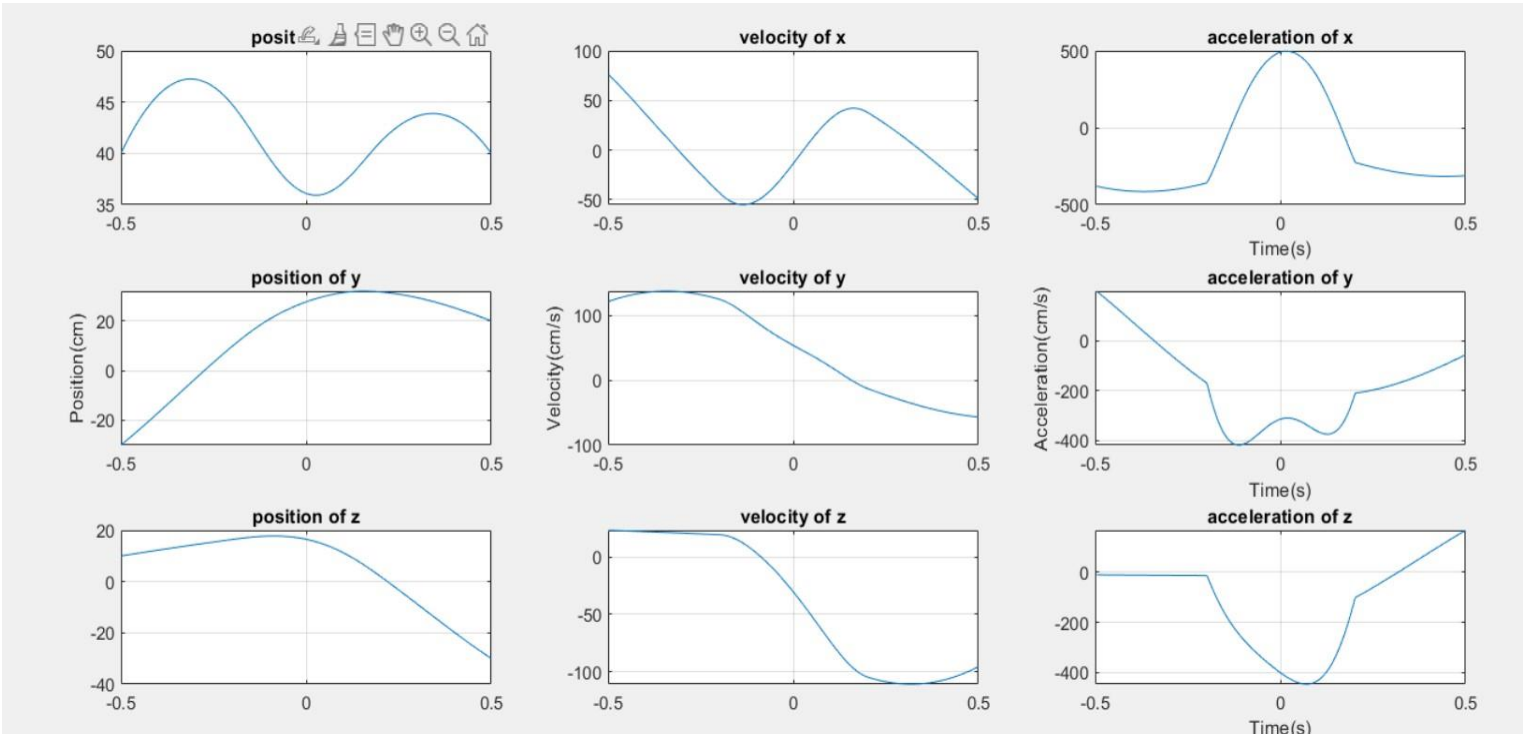
$h = \frac{t - t_1}{t_3 - t_1}$

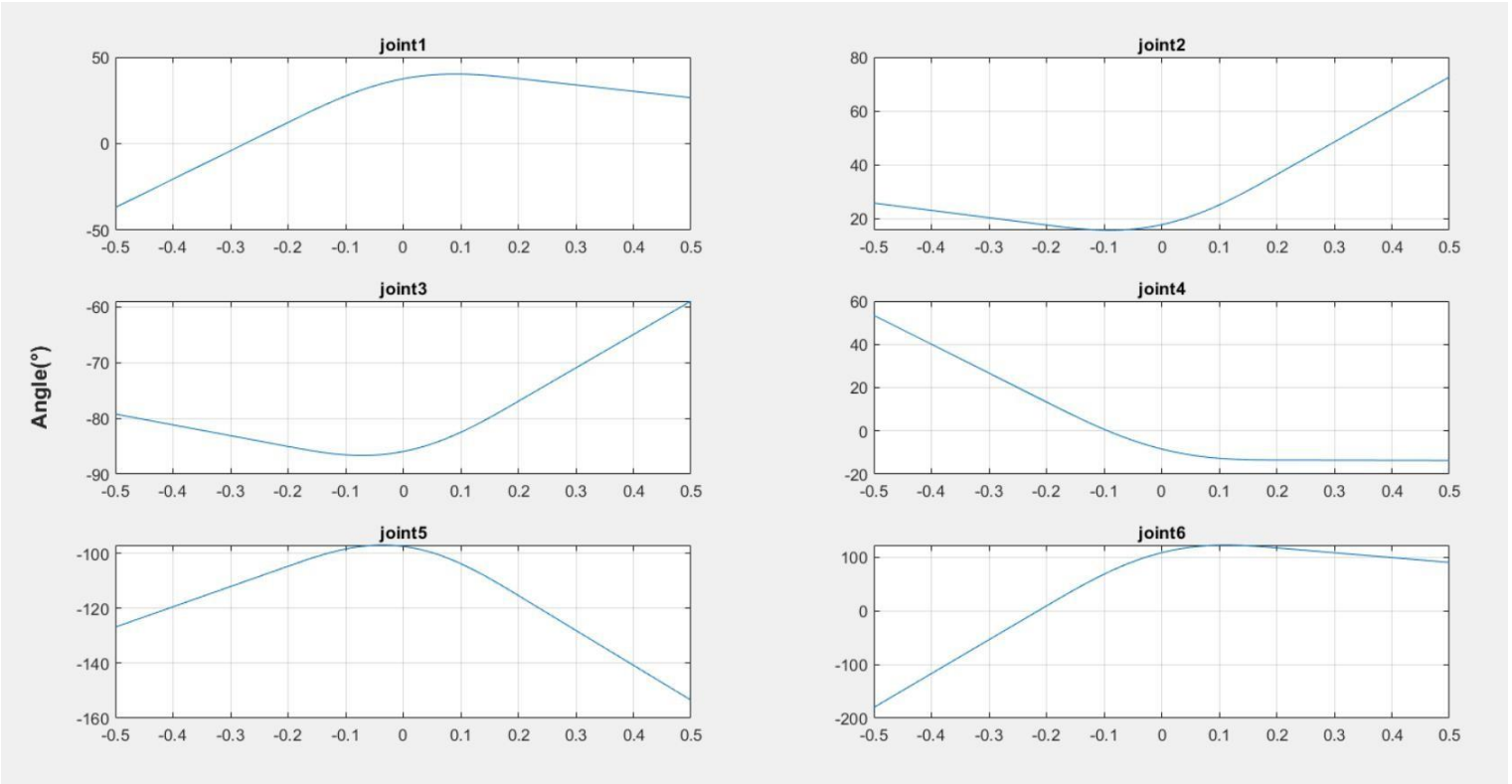
angle: $\left(\left(\Delta C \cdot \frac{t_{acc}}{t_2} - \Delta B \cdot \frac{t_{acc}}{t_2} \right)(2-h)h^2 + 2\Delta B \cdot \frac{t_{acc}}{t_2} \right) h + B - \Delta B \frac{t_{acc}}{t_2}$

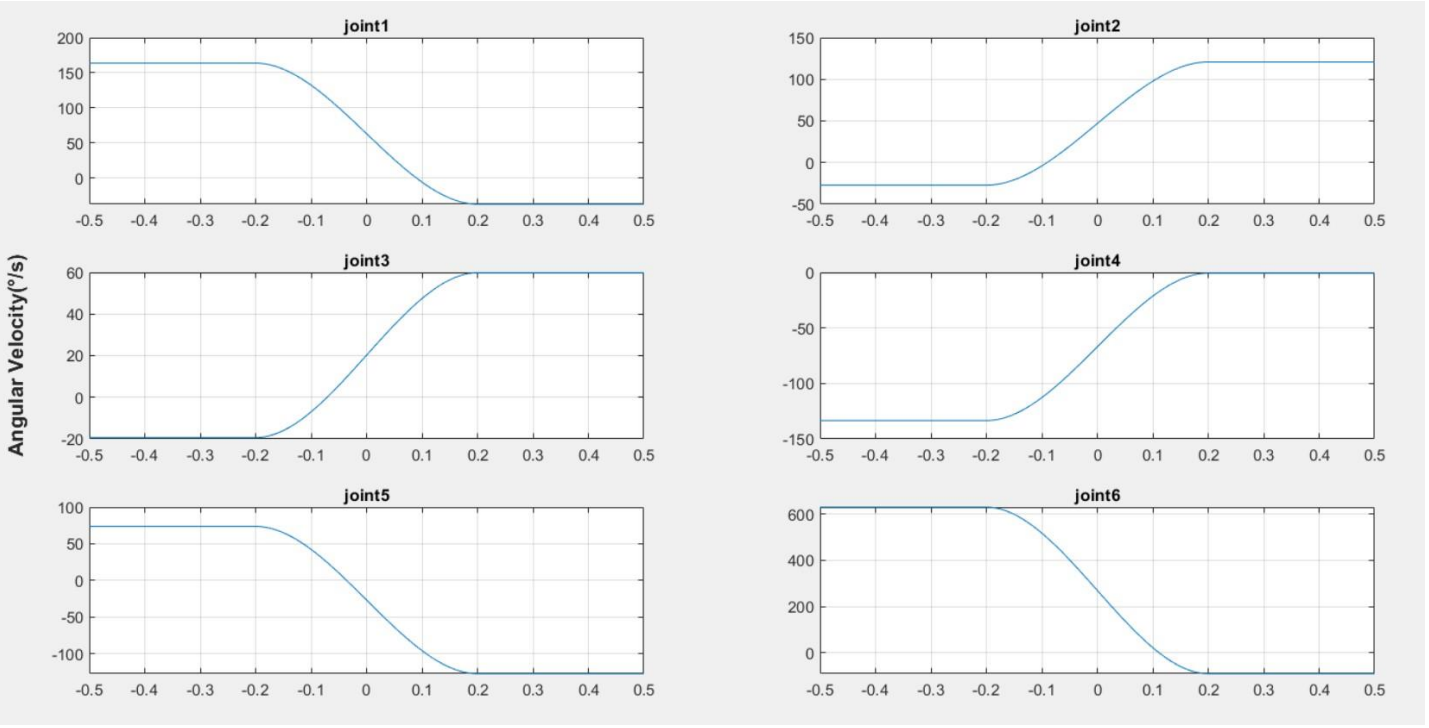
angle_vel: $\left(\left(\Delta C \cdot \frac{t_{acc}}{t_2} - \Delta B \cdot \frac{t_{acc}}{t_2} \right) \cdot (1.5-h) \cdot 2h^2 + \Delta B \frac{t_{acc}}{t_2} \right) \cdot \frac{1}{t_{acc}}$

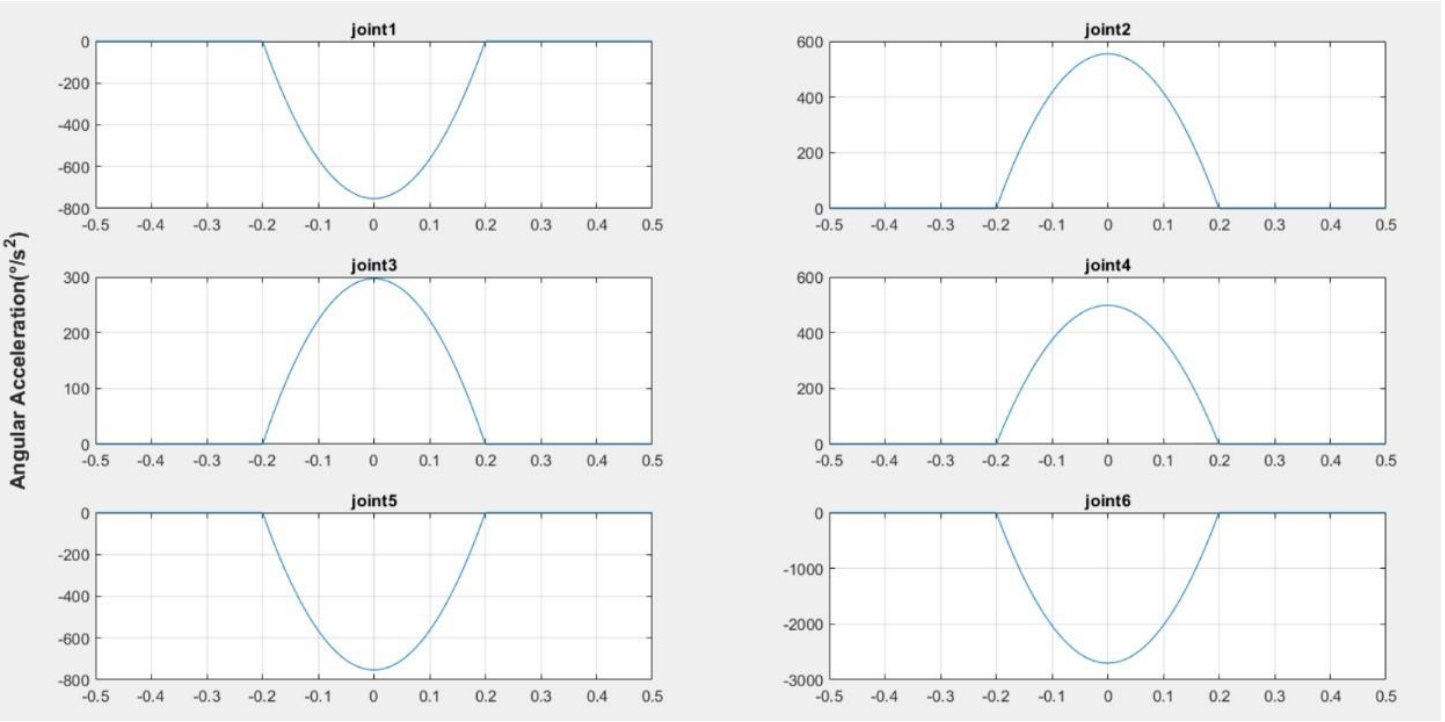
angle_acc: $\left(\Delta C \frac{t_{acc}}{t_2} - \Delta B \cdot \frac{t_{acc}}{t_2} \right) \cdot (1-h) \cdot \frac{3h}{t_{acc}^2}$

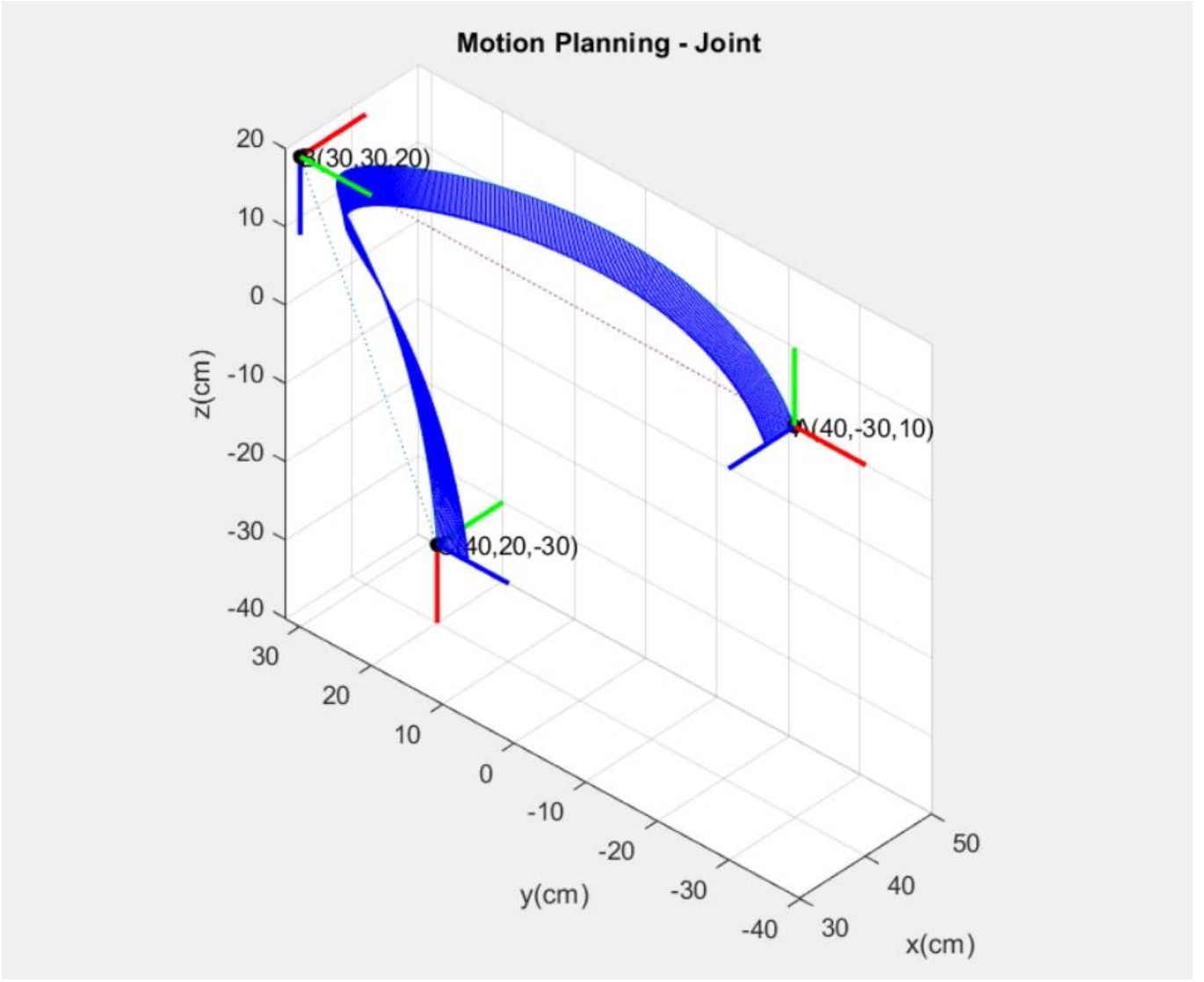
Result : Joint Move



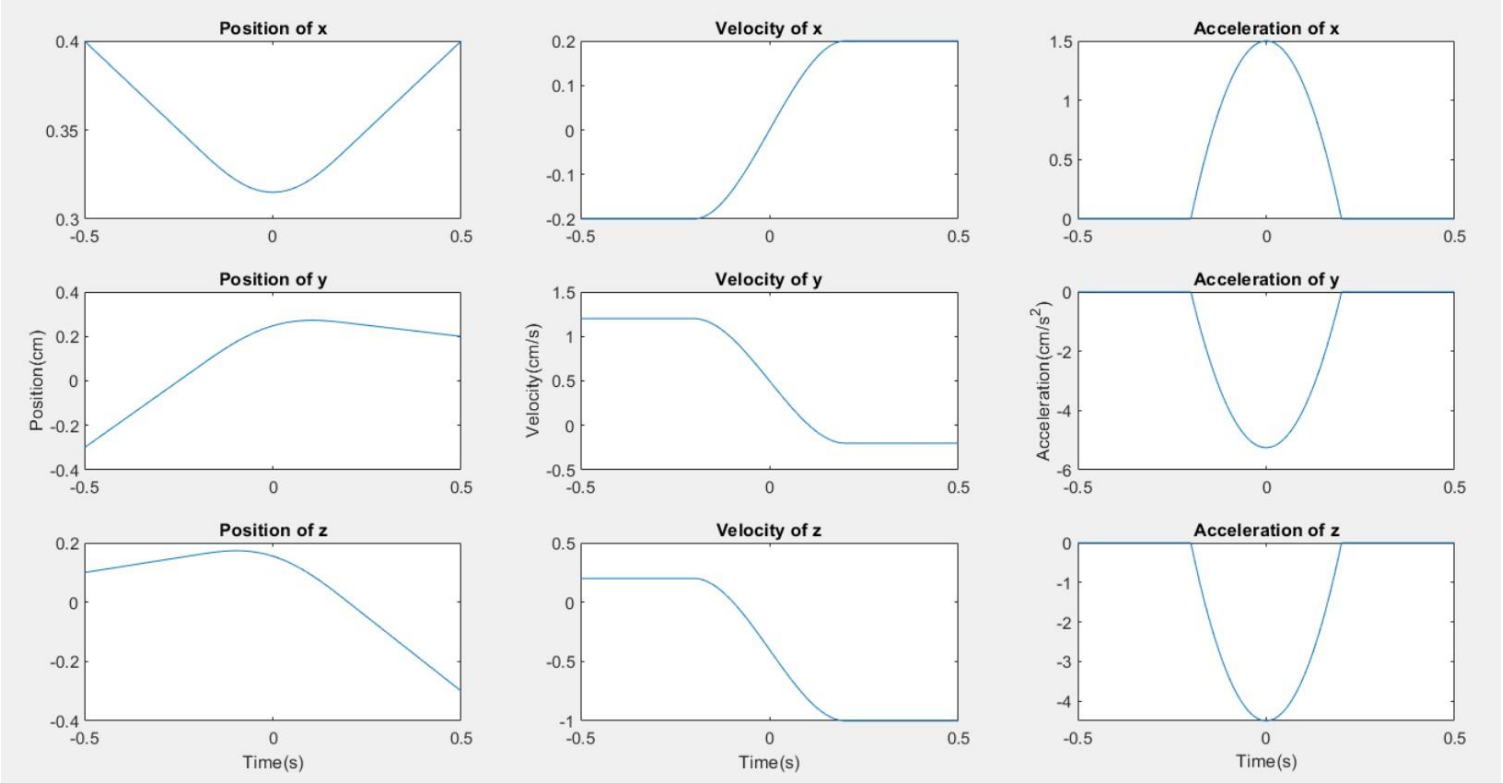


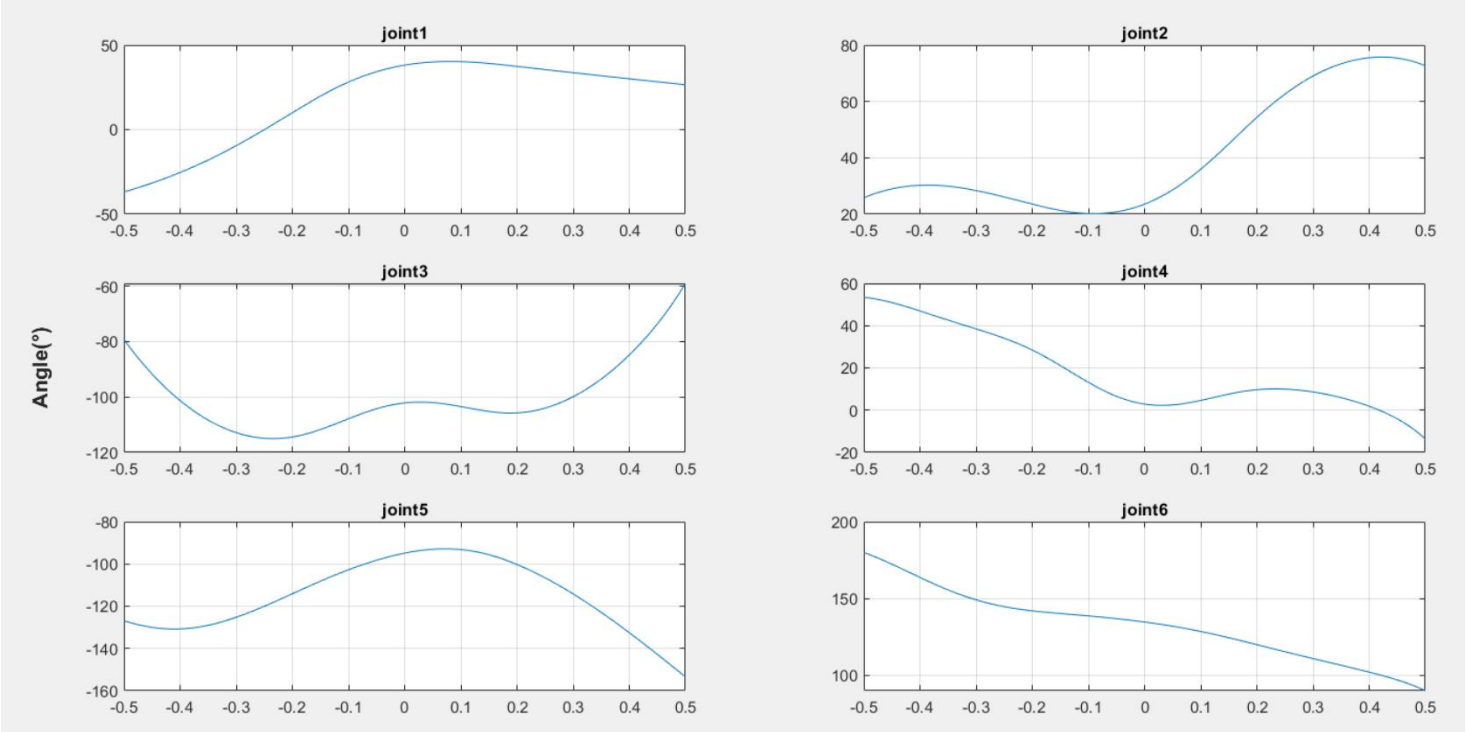

joint1
joint2
joint3
joint4
joint5
joint6
Angular Velocity(°/s)

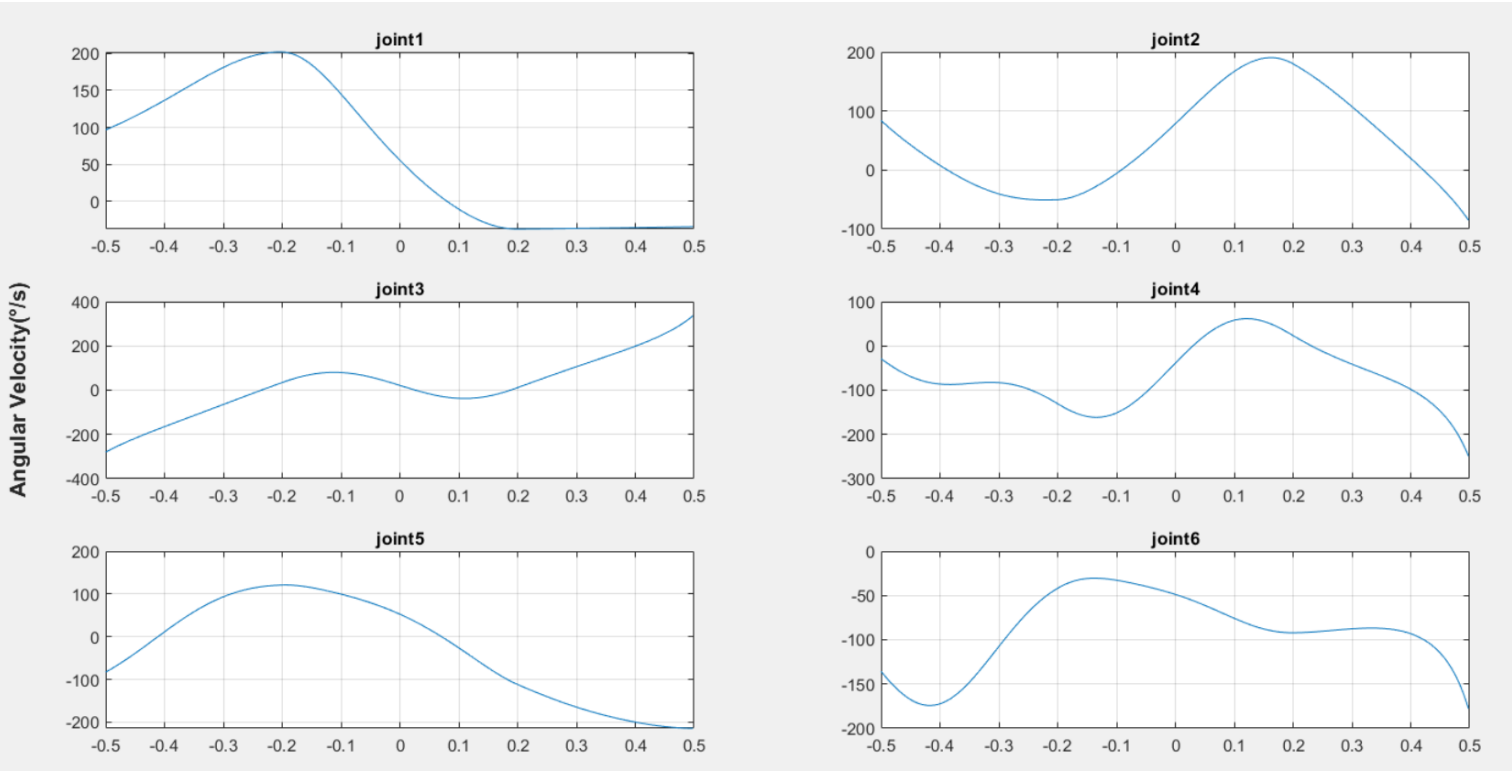

joint1
joint2
joint3
joint4
joint5
joint6
Angular Acceleration(°/s^2)

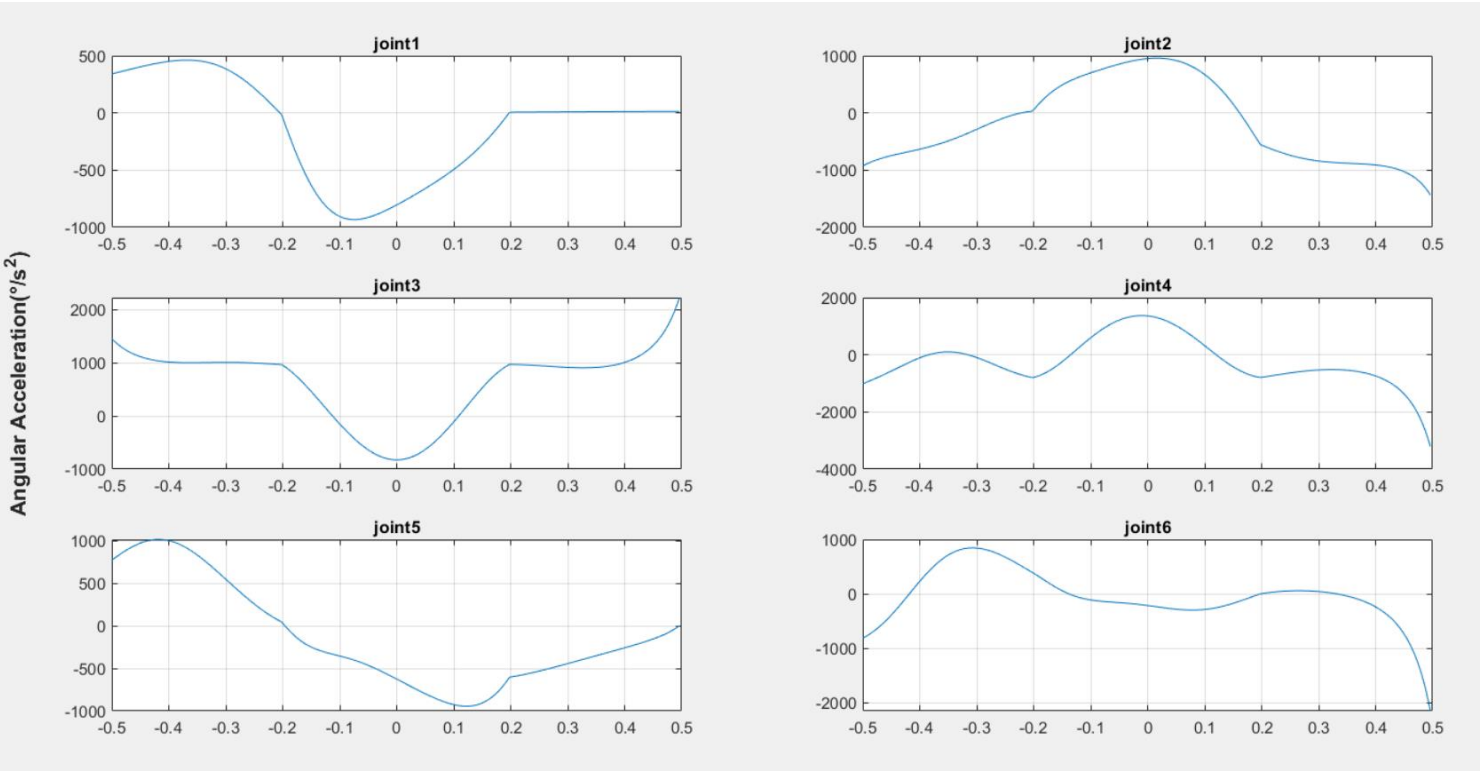

Motion Planning - Joint
20
10
0
-10
-20
-30
-40
z(cm)
30
20
10
0
-10
-20
-30
-40
y(cm)
50
40
30
x(cm)
(30,30,20)
(40,-30,10)
40,20,-30)

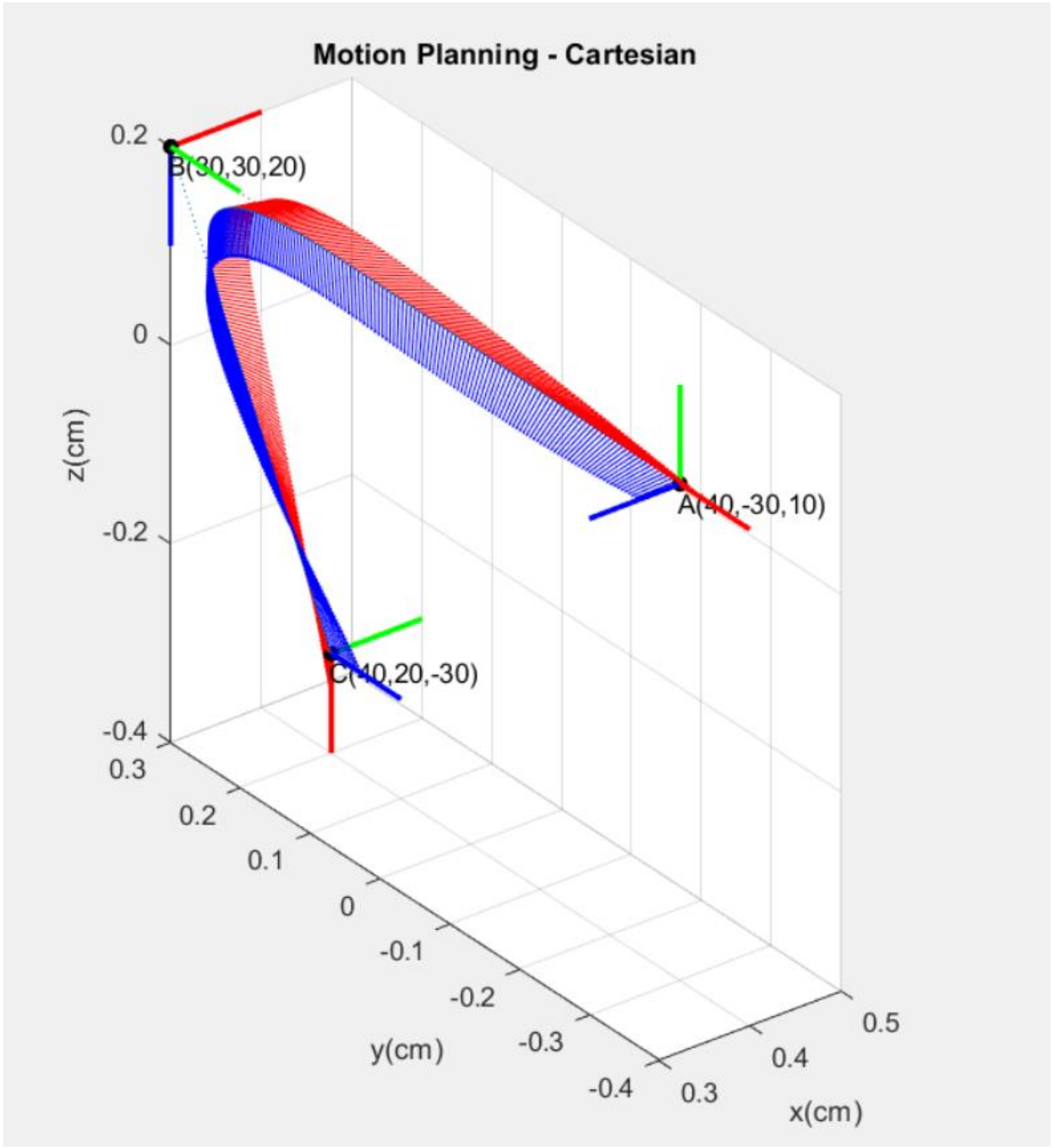
Result: Cartesian Move






Angular Velocity(°/s)
joint1
joint2
joint3
joint4
joint5
joint6


Angular Acceleration(°/s^2)
joint1
joint2
joint3
joint4
joint5
joint6


Motion Planning - Cartesian
0.2
B(30,30,20)
0
-0.2
-0.4
z(cm)
A(40,-30,10)
C(40,20,-30)
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
y(cm)
0.5
0.4
0.3
x(cm)

III. 討論 兩種軌跡優缺點

Joint Move

A. 優點

1. 只需做端點的 IK，運算量大

2. 馬達運動比較平順

3. 不需要考慮奇異點 (In joint space)

B. 缺點

1. 路徑比 Cartesian 扭曲

2. 不直觀 (Joint Space)

Cartesian Move

A. 優點

1. 路徑不曲折

2. 做規劃時比較直覺 (卡式坐標系)

B. 缺點

1. 運算量大 (每個路徑上的點都要算 IK)

2. 馬達運動比較不平順

3. 要考慮奇異點