به نام خدا

تمرین شماره ۱

درس کنترل سیستم های عصبی عضلانی

تهیه کننده: علیرضا امیری

شماره دانشجویی: ۴۰۲۰۲۴۱۴

استاد درس: دکتر دلربایی

پاییز ۱۴۰۲

۱. کاربرد های عملی MJT در دنیای واقعی

فعالیت های روزمره:

۱. *نوشتن*: در فرایند نوشتن، دست انسان با استفاده از MJT، مسیری هموار و بدون تنش را طی می کند و در نتیجه، خطوط رسم شده بر روی کاغذ بسیار طبیعی به نظر می رسند.
۲. *راه رفتن*: به هنگام راه رفتن، پا به گونه ای حرکت می‌کند و تاب می‌خورد که مسیر طی شده دارای کمترین جرک باشد.
۳. *غذا خوردن*: برای برداشتن غذا از روی میز و رساندن قاشق به دهان، معمولا دست انسان مسیری با کمینه جرک طی می‌کند.
۴. ساز زدن: در هنگام ساز زدن، معمولا انگشتان نوازنده برای جابه‌جایی بین موقعیت های مختلف، مسیری با کمینه جرک را طی می‌کند.

کاربردهای عملی MJT:

۱. بازو های رباتیک: در بازو های رباتیک، قسمت ابزار بازو باید مسیر بین نقاط مختلف را طی کند. بنابراین، با اعمال MJT، می‌توان مسیری بازو را طوری طراحی کرد که برای جابه‌جایی بین نقاط مختلف، مسیری بهینه و هموار را طی کند.
۲. سیستم های همکاری انسان و ربات (human-robot cooperation):
در این سیستم ها، از آنجایی که بازوی رباتی باید بتواند کاملا با حرکت دست انسان هماهنگ باشد، باید قادر باشد مانند دست انسان، مسیری هموار و بدون تنش را طی کند. بنابراین، در این سیستم ها از MJT استفاده می‌شود.

۳. کاربرد های لمسی (Haptic Application):

در فرایند توان بخشی، برای بیمارانی که بر اثر سکته توانایی کنترل بخشی از ماهیچه های دست و پای خود را از دست داده‌اند، از MJT به منظور طراحی حرکت های مانند انسان (human-like) بهره برده می‌شود.

۲. رابطه‌ی موقعیت و سرعت و MJT در حرکت خطی مستقیم

Minimum Jerk Trajectory $\rightarrow M = \int_0^{t_f} \|\dddot{x}\|^2 dt$

$\rightarrow$ در حرکت خطی مستقیم $\Rightarrow \begin{cases} x(t) = x_0 + (x_f - x_0)(6\tau^5 - 15\tau^4 + 10\tau^3) \\ y(t) = y_0 + (y_f - y_0)(6\tau^5 - 15\tau^4 + 10\tau^3) \end{cases}$ [رابطه‌ی موقعیت]

$\begin{cases} \tau \rightarrow & \text{زمان نرمالیز شده} \Rightarrow 0 \leq \tau \leq 1 \\ x_0, y_0 \rightarrow & \text{نقطه شروع حرکت} \\ x_f, y_f \rightarrow & \text{نقطه پایان حرکت} \end{cases}$

$$v_{Jerk\,x} = \dot{x}(t) = x_f(30\tau^4 - 60\tau^3 + 30\tau^2)$$

$$v_{Jerk\,y} = \dot{y}(t) = y_f(30\tau^4 - 60\tau^3 + 30\tau^2)$$

۳. رابطه‌ی موقعیت و سرعت و MMJT در حرکت منحنی‌الخط

$$\boxed{x^-(\tau)} = \frac{t_f^5}{720}\Big(\pi_m\big(\tau_m^4(15\tau^4-30\tau^3)+\tau_m^3(80\tau^3-30\tau^4)-60\tau^3\tau_m^2+30\tau^4\tau_m-6\tau^5\big)+c_m(15\tau^4-10\tau^3-6\tau^5)\Big)+x_0 \quad (1)$$

$$\boxed{x^+(\tau)} = x^-(\tau)+\pi_m\frac{\tau_m^5(\tau-\tau_m)^5}{120} \quad (2)$$

$$\tau_m = \frac{t_m}{t_f} \qquad \tau = \frac{t}{t_f}$$

t_m در نظری ⇒ $\boxed{x^-(t_m) = x^+(t_m) = x_m}$ I

→ جایگذاری I در روابط ۱و۲ ⇒ ساده‌سازی ⇒

حل برای متغیرهای π_m , c_m

$$c_m = \frac{1}{t_f^5\tau_m^2(1-\tau_m)^5}\underbrace{\Big((x_f-x_0)(300\tau_m^5-1200\tau_m^4+1600\tau_m^3)+\tau_m^2(-720x_f+120x_m+600x_0)+(x_0-x_1)(300\tau_1-200)\Big)}_{A_1}$$

$$\Rightarrow \boxed{c_m} = \frac{1}{t_f^5\tau_m^2(1-\tau_m)^5}A_1$$

$$\pi_1 = \frac{1}{t_f^5\tau_m^2(1-\tau_m)^5}\underbrace{\Big((x_f-x_0)(120\tau_m^5-300\tau_m^4+200\tau_m^3)-20(x_m-x_0)\Big)}_{A_2}$$

$$\boxed{\pi_1} = \frac{1}{t_f^5\tau_m^2(1-\tau_m)^5}A_2$$

⇒ مشتق از روابط موقعیت ⇒ سرعت ⟶

$$\boxed{\mathcal{U}^{-}_{jerk\,x}} = \frac{t_f^{5}}{720}\Big(\pi_m\big(\tau_m^{4}(60\tau^{3}-90\tau^{2})+\tau_m^{3}(240\tau^{2}-120\tau^{3})$$

$$-180\tau^{2}\tau_m^{2}+120\tau^{3}\tau_m-30\tau^{4}\big)+C_m(60\tau^{3}-30\tau^{2}-30\tau^{4})\Big)$$

$$\boxed{\mathcal{U}^{+}_{jerk\,x}} = \mathcal{U}^{-}_{jerk\,x}+5\pi_m\frac{\tau_m^{5}(\tau-\tau_m)^{4}}{120} = \mathcal{U}^{-}_{jerk}+\pi_m\frac{\tau_m^{5}(\tau-\tau_m)^{4}}{24}$$

روابط مشابه برای سرعت و موقعیت در راستای y نیز با تعریف y^{+} ، y^{-} ، π_n ، τ_n ، $\mathcal{U}^{-}_{jerk\,y}$ ، $\mathcal{U}^{+}_{jerk\,y}$ برقرار می باشد. همچنین مقادیر A_1' و A_2' نیز متقابلاً تعریف می شود.

حال باید مقدار π_m را بدست آوریم.

طبق رابطه‌ی همیلتون $\rightarrow \pi_m u(t_m) + \pi_n v(t_m) = 0$

که داریم: $\begin{cases} u(t_m) = \dot{x}_m \\ u(t_m) = \dot{y}_m \end{cases}$ در لحظه‌ی t_m $\rightarrow \dot{x}^{+}(t_m) = \dot{x}^{-}(t_m) = \dot{x}_m$

$\dot{y}^{+}(t_m) = \dot{y}^{-}(t_m) = \dot{y}_m$

$$\Rightarrow \pi_1\dot{x}_m + \pi_2\dot{y}_m = 0 \quad \mathrm{II}$$

$$\Rightarrow \mathring{x}_m = \frac{t_f^5}{720}\left(\pi_m\left(\tau_m^4(60\tau_m^3 - 90\tau_m^2) + \tau_m^3(240\tau_m^2 - 120\tau_m^3) - 180\tau_m^4 + 120\tau_m^4 - 30\tau_1^4\right) + C_m(60\tau_m^3 - 30\tau_m^2 - 30\tau_m^4)\right) =$$

$$= \frac{t_f^5}{720}\left(\pi_m(60\tau_m^7 - 90\tau_m^6 + 240\tau_m^5 - 120\tau_m^6 - 90\tau_m^4) + C_m(60\tau_m^3 - 30\tau_m^2 - 30\tau_m^4)\right)$$

$$= \frac{t_f^5}{720}\left(\pi_m\underbrace{(60\tau_m^7 - 210\tau_m^6 + 240\tau_m^5 - 90\tau_m^4)}_{A_3} + C_m\underbrace{(60\tau_m^3 - 30\tau_m^2 - 30\tau_m^4)}_{A_4}\right)$$

$$\frac{t_f^5}{720}(\pi_m A_3 + C_m A_4)$$

به طریق مشابه:

$$\mathring{y}_m = \frac{t_f^5}{720}(\pi_n A_3 + C_n A_4)$$

$\longrightarrow$ II جایگذاری در $\Rightarrow$

$$\pi_m\left(\cancel{\frac{t_f^5}{720}}(\pi_m A_3 + C_m A_4)\right) + \pi_n\left(\cancel{\frac{t_f^5}{720}}(\pi_n A_3 + C_n A_4)\right) = 0$$

$$\Rightarrow \pi_m^2 A_3 + \pi_m C_m A_4 + \pi_n^2 A_3 + \pi_n C_n A_4 = 0$$

$$\Rightarrow A_3(\pi_m^2 + \pi_n^2) + A_4(\pi_m C_m + \pi_n C_n) = 0$$

جایگذاری روابط به دست آمده برای π_{2m}، π_{2n}، C_m و C_n در رابطه‌ی اخیر

$$A_2\left(\left(\frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A_2\right)^2 + \left(\frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A'_2\right)^2\right)$$

$$+ A_4\left(\frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A_2 \times \frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A_1\right.$$

$$\left. + \frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A'_2 \times \frac{1}{tf^5 Z_m^5 (1-Z_m)^5} A'_1\right) = 0$$

$$\Rightarrow A_2\left(\left(\frac{1}{Z_m^5} A_2\right)^2 + \left(\frac{1}{Z_m^5} A'_2\right)^2\right) + A_4\left(\frac{1}{Z_m^5} A_2 \frac{1}{Z_m^2} A_1 + \frac{1}{Z_m^5} A'_2 \frac{1}{Z_m^2} A'_1\right) = 0$$

$$A_2\left(\frac{1}{Z_m^{10}} A_2^2 + \frac{1}{Z_m^{10}} A'^2_2\right) + A_4\left(\frac{1}{Z_m^7} A_1 A_2 + \frac{1}{Z_m^7} A'_1 A'_2\right) = 0$$

$$\boxed{A_2 A_2^2 + A_2 A'^2_2 + Z_m^3 A_1 A_2 A_4 + Z_m^3 A'_1 A'_2 A'_4 = 0} \rightarrow$$

جایگذاری مقادیر A و حل معادله برای به دست آوردن Z_m

کد متلب برای حل معادله ی فوق:

```
clc
clear all;
syms t1 tf xf yf x1 y1 float
t1 = sym('t1', 'real');
tf = sym('tf', 'real');
xf = sym('xf', 'real');
yf = sym('yf', 'real');
x1 = sym('x1', 'real');
y1 = sym('y1', 'real');
xf = 0.6
yf = 0.0
x1 = 0.1
y1 = 0.3
%%
A2 = expand(xf*(120*t1^5 - 300*t1^4 + 200*t1^3) -20*x1)
A22 = expand(yf*(120*t1^5 - 300*t1^4 + 200*t1^3) -20*y1)
A1 = expand(xf*(300*t1^5 - 1200*t1^4 + 1600*t1^3) + t1^2*(-720*xf + 120*x1) -
x1*(300*t1 - 200))
A11 = expand(yf*(300*t1^5 - 1200*t1^4 + 1600*t1^3) + t1^2*(-720*yf + 120*y1) -
y1*(300*t1 - 200))
A3 = 60*t1^7 - 210*t1^6 + 240*t1^5 - 90*t1^4
A4 = 60*t1^3 - 30*t1^2 - 30*t1^4
Result = expand(A3*A2*A2 + A3*A22*A22 + (t1^3)*A1*A2*A4 + (t1^3)*A11*A22*A4)
%numerical solution
solvet = vpasolve(Result, t1)
```

پس از حل معادله به وسیله‌ی متلب و پیدا شدن ریشه های معادله، تنها ریشه‌ای را می‌پذیریم که مقدار آن نرمالایز باشد، یعنی عددی میان ۰ و ۱.

```
solvet =

                                     -0.57512787896041684267598344706099
                                                                       0
                                                                       0
                                                                       0
                                                                       0
                                      0.43554096967461129711909085705565
                                                                     1.0
                                                                     1.0
                                                                     1.0
                                                                     1.0
                                       1.9380928639462514599232686417604
- 0.24289662098861503203466292370776 + 0.34729014652581125310253545256349i
- 0.24289662098861503203466292370776 - 0.34729014652581125310253545256349i
  0.33675578452905668748920376262455 + 0.44792321386042112215626099493977i
  0.33675578452905668748920376262455 - 0.44792321386042112215626099493977i
   1.2568878591293353873622711352057 + 0.33305752349539549316127717781801i
   1.2568878591293353873622711352057 - 0.33305752349539549316127717781801i
```

در این مثال، با در نظر گرفتن مقادیر اولیه‌ی مفروض، تنها مقدار ۰/۴۳۵ قابل قبول می‌باشد.

برای انتخاب عدد حقیقی که میان دو مقدار ۰ و ۱ باشد، به وسیله ی دو دستور زیر ریشه ها را فیلتر می‌کنیم.

```
real_roots = solvet(imag(solvet)==0); % filter out only real roots
T1 = real_roots(real_roots > 0 & real_roots < 1) % filter out roots between 0 and 1
```

در نهایت، ریشه‌ی مورد نظر در متغیر T1 ذخیره می‌شود.

```
T1 =

0.43554096967461129711909085705565
```

- رسم نمودار های موقعیت و سرعت در راستا های X و Y

با مشخص شدن زمان رسیدن به نقطه‌ی میانی، می‌توانیم از تابع های موقعیت و سرعت جسم استفاده کرده و با جایگذاری مقادیر T1، در هر زمان t موقعیت و سرعت آن را به دست آوریم. رابطه های مورد نیاز در این قسمت، در محاسبات بالا به با مستطیل قرمز مشخص شده اند.

در گام اول، این روابط در محیط متلب نوشته می‌شوند.

۱. روابط مربوط به موقعیت:

```
%% X-Position Equations as a function of time

pi1 = 1/(tf^5 * t1^5 * ((1-t1)^5)) * A2;
c1 = 1/(tf^5 * t1^2 * ((1-t1)^5)) * A1;
star1(t) = t1^4 * (15*t^4 - 30*t^3)+t1^3 * (80*t^3-30*t^4)-60*t^3*t1^2+30*t^4*t1-6*t^5;
star2(t) = 15*t^4 - 10*t^3 - 6*t^5;

 x_negative(t,t1) = (tf^5)/720 * ((pi1*star1)+ c1 *star2);
 x_positive(t,t1) =x_negative(t,t1) + pi1* (t1^5 * ((t-t1)^5))/120;

Position_X(t,t1) =  heaviside(t1 - t) * x_negative(t,t1) +heaviside(t - t1) * x_positive(t,t1) ;

%% Y-Position Equations as a function of time
pi2 = 1/(tf^5 * t1^5 * ((1-t1)^5)) * A22;
c2 = 1/(tf^5 * t1^2 * ((1-t1)^5)) * A11;

 y_negative(t,t1) = (tf^5)/720 * ((pi2*star1)+ c2 *star2);
 y_positive(t,t1) = y_negative(t,t1) + pi2* (t1^5 * ((t-t1)^5))/120;

Position_Y(t,t1) =  heaviside(t1 - t) * y_negative(t,t1) +heaviside(t - t1) * y_positive(t,t1) ;
```

۲. روابط مربوط به سرعت:

```
%% X-Velocity Equations as a function of time

Vx_negative = tf^5/720*(pi1*(t1^4*(60*t^3-90*t^2)+t1^3*(240*t^2-120*t^3)-180*t^2*t1^2+120*t^3*t1-30*t^4)+c1*(60*t^3-30*t^2-30*t^4));
Vx_positive = Vx_negative + pi1/24*(t1^5*(t-t1)^4);

Velocity_X(t,t1) = heaviside(t1 - t) * Vx_negative + heaviside(t - t1) * Vx_positive;

%% Y-Velocity Equations as a function of time
```

```
Vy_negative = tf^5/720*(pi2*(t1^4*(60*t^3-90*t^2)+t1^3*(240*t^2-120*t^3)-
180*t^2*t1^2+120*t^3*t1-30*t^4)+c2*(60*t^3-30*t^2-30*t^4));
Vy_positive = Vy_negative + pi2/24*(t1^5*(t-t1)^4);

Velocity_Y(t,t1) = heaviside(t1 - t) * Vy_negative + heaviside(t - t1) * Vy_positive;
```

در نهایت، هر یک از نمودار های مورد نظر به وسیله‌ی دستور fplot در بازه‌ی [0,tf] رسم می‌شوند.

۴. الف) حرکت در راستای y

X0 = y0 = xf = 0

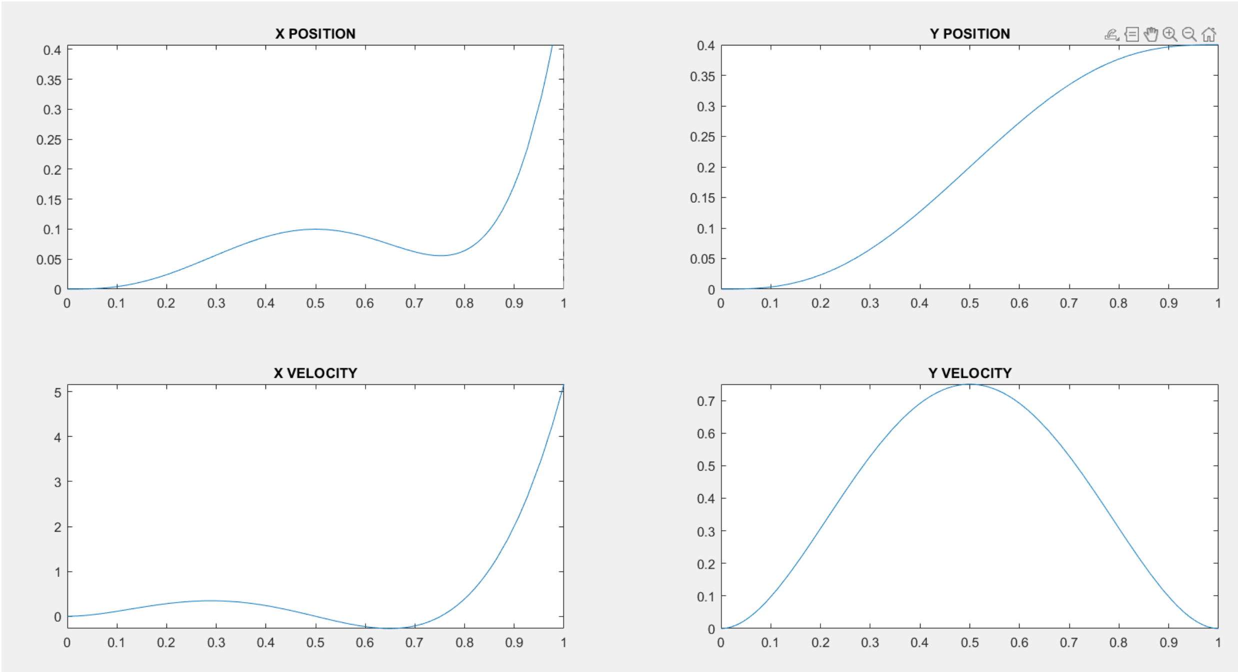
Yf = 0.4m

xm = 0.1m

ym = 0.2m

با جایگذاری مقادیر فوق در کد نوشته شده، نتایج زیر حاصل می‌شود.

```
solvet =

                                      -0.77417897713215564805124909729923
                                                                        0
                                                                        0
                                                                        0
                                                                        0
                                                                      0.5
                                                                      1.0
                                                                      1.0
                                                                      1.0
                                                                      1.0
                                        1.7741789771321556480512490972992
- 0.21148640677679780791693237907037 - 0.28314788352696468424535819947396i
- 0.21148640677679780791693237907037 + 0.28314788352696468424535819947396i
                                 0.5 + 0.37719901722689041272894359624186i
                                 0.5 - 0.37719901722689041272894359624186i
   1.2114864067767978079169323790704 + 0.28314788352696468424535819947396i
   1.2114864067767978079169323790704 - 0.28314788352696468424535819947396i


T1 =

0.5
```



۴. ب) حرکت در راستای x

X0 = y0 = yf = 0

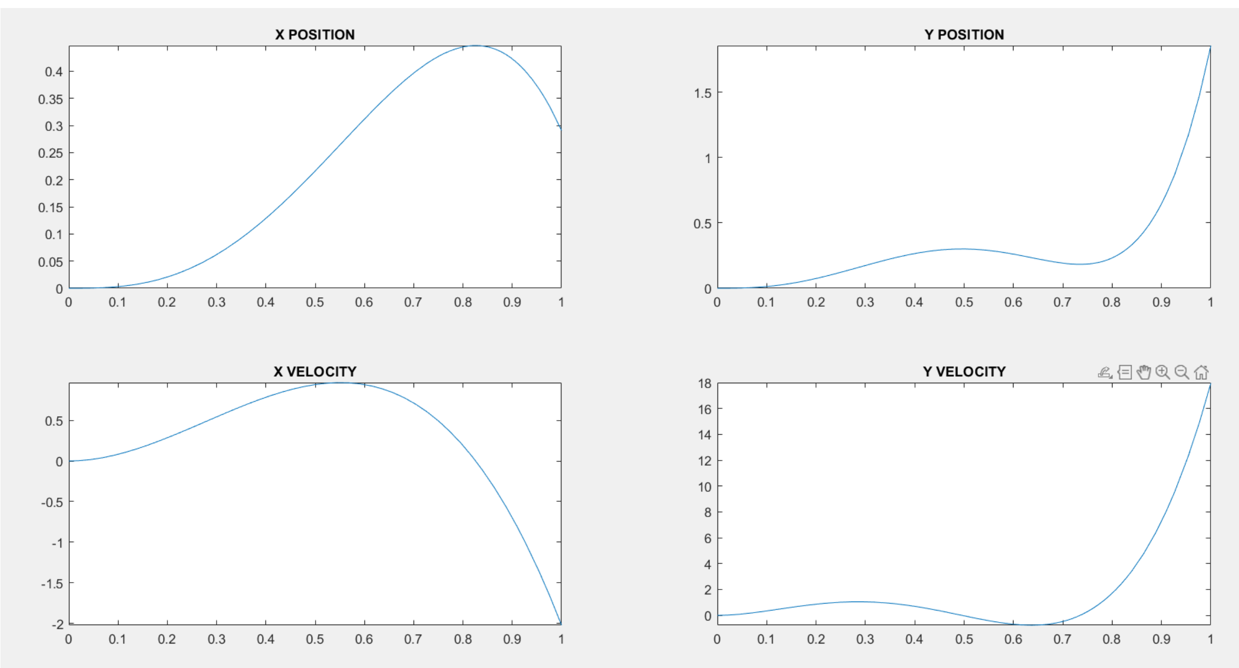
xf = 0.5m

xm = 0.2m

ym = 0.3m

مجدد، با جایگذاری مقادیر فوق در کد، نتایج زیر حاصل می‌شود:

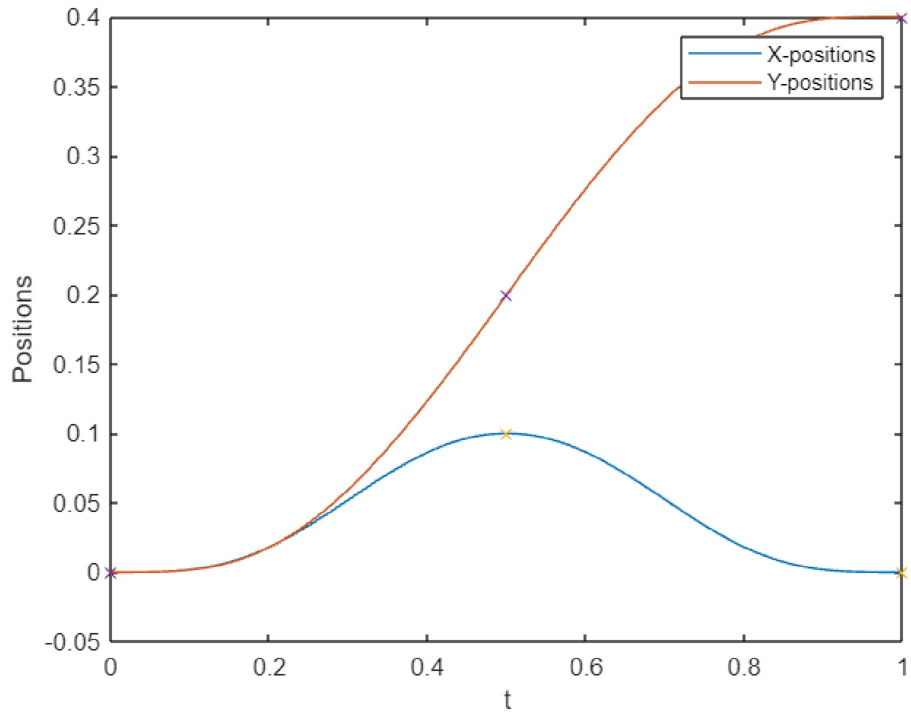
```
solvet =

                                              -0.73285862860441994579867081893747
                                                                                0
                                                                                0
                                                                                0
                                                                                0
                                               0.48268762708301056243658445368698
                                                                              1.0
                                                                              1.0
                                                                              1.0
                                                                              1.0
                                                   1.83698969496650971799852538029
- 0.25314697800326521648181235139218 - 0.34070568354064983626452482690371i
- 0.25314697800326521648181235139218 + 0.34070568354064983626452482690371i
  0.45381270028046211021342133361193 - 0.48156128201161020045768970000235i
  0.45381270028046211021342133361193 + 0.48156128201161020045768970000235i
   1.2559249310002529389501715102605 + 0.33669267053083359011771018144566i
   1.2559249310002529389501715102605 - 0.33669267053083359011771018144566i


T1 =

0.48268762708301056243658445368698
```



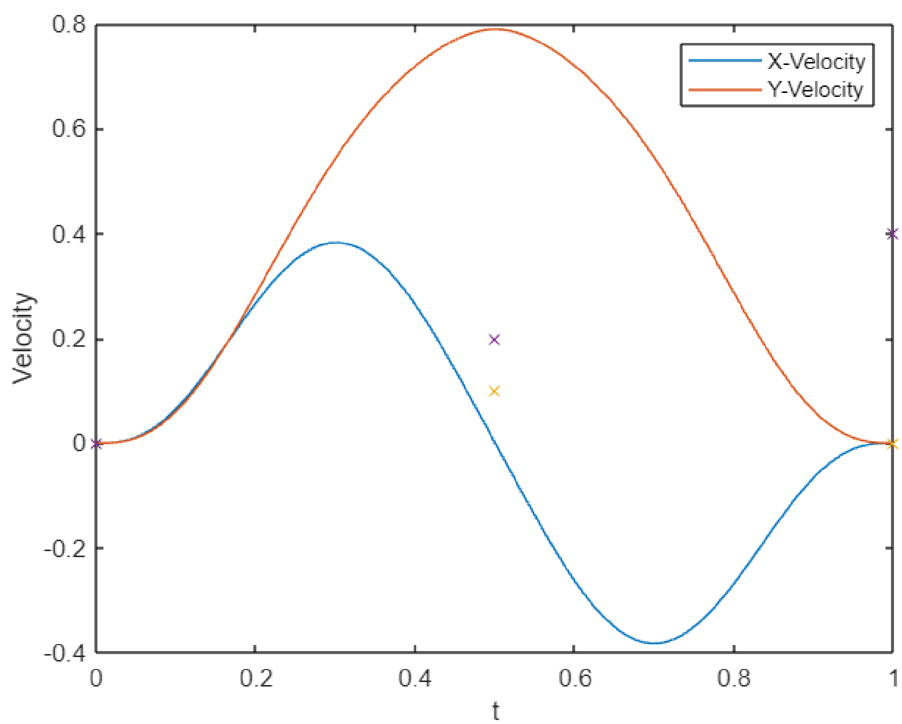
روش دوم حل (با استفاده از ابزار minjerkpolytraj متلب)

۱. حرکت در راستای Y

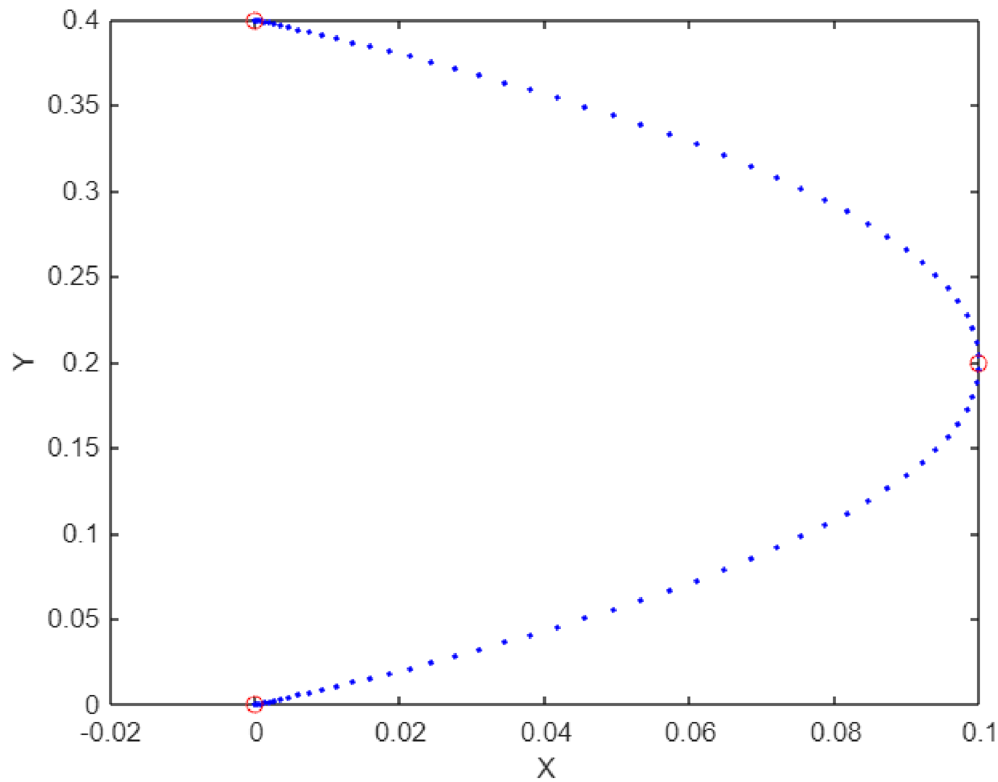
```
wpts = [0 0.1 0; 0 0.2 0.4];
tpts = 0:0.5:1;
numsamples = 100;

[q,qd,qdd,qddd,pp,timepoints,tsamples] =
minjerkpolytraj(wpts,tpts,numsamples);

plot(tsamples,q)
hold on
plot(timepoints,wpts,'x')
xlabel('t')
ylabel('Positions')
legend('X-positions','Y-positions')
hold off

figure
plot(tsamples,qd)
hold on
plot(timepoints,wpts,'x')
xlabel('t')
ylabel('Velocity')
legend('X-Velocity','Y-Velocity')
hold off


figure
plot(q(1,:),q(2,:),'.b',wpts(1,:),wpts(2,:),'or')
xlabel('X')
ylabel('Y')
```



X-positions
Y-positions
Positions
t
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
0
0.2
0.4
0.6
0.8
1



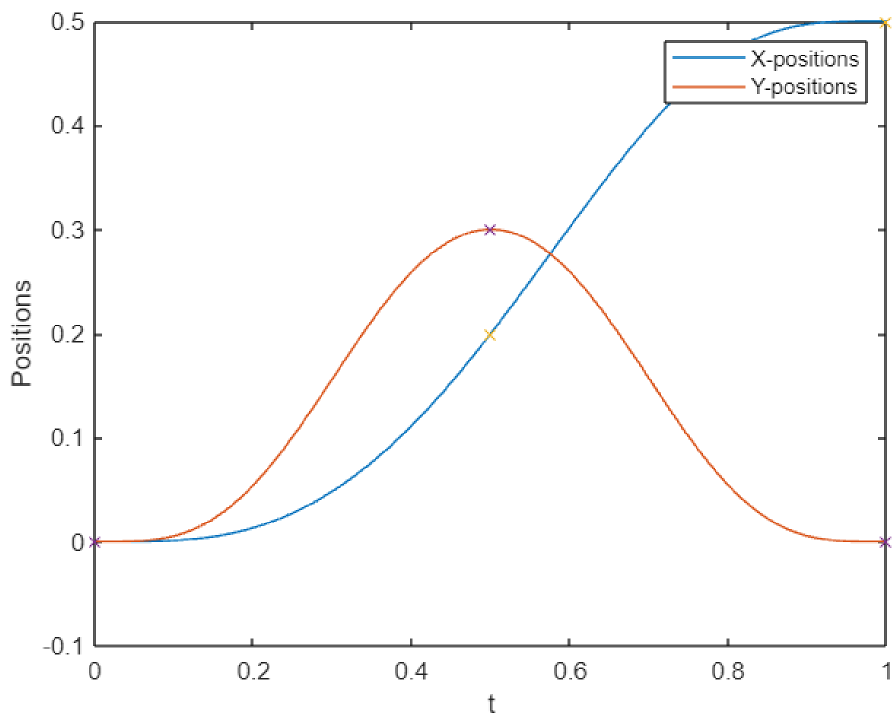
X-Velocity
Y-Velocity
Velocity
t
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
0.2
0.4
0.6
0.8
1

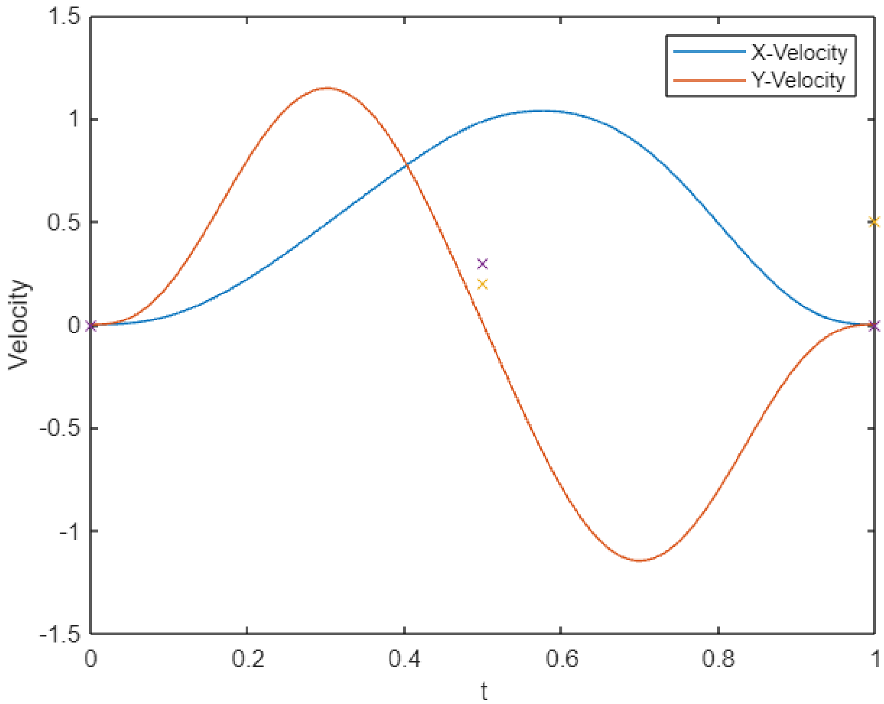


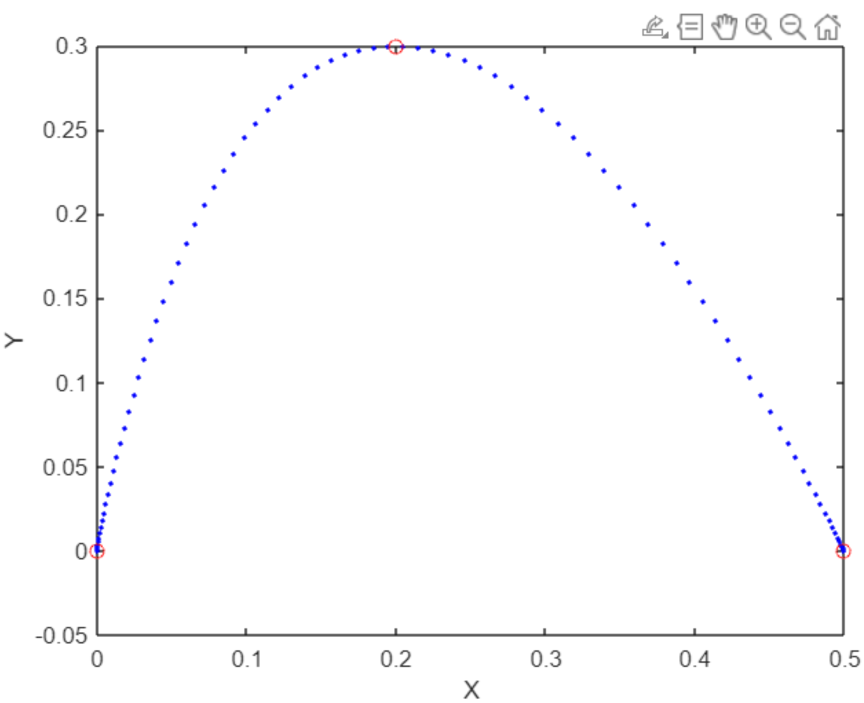
۲. حرکت در راستای X

```
wpts = [0 0.2 0.5; 0 0.3 0];
tpts = 0:0.5:1;
numsamples = 100;


[q,qd,qdd,qddd,pp,timepoints,tsamples] = minjerkpolytraj(wpts,tpts,numsamples);


plot(tsamples,q)
hold on
plot(timepoints,wpts,'x')
xlabel('t')
ylabel('Positions')
legend('X-positions','Y-positions')
hold off


figure
plot(tsamples,qd)
hold on
```

```
plot(timepoints,wpts,'x')
xlabel('t')
ylabel('Velocity')
legend('X-Velocity','Y-Velocity')
hold off



figure
plot(q(1,:),q(2,:),'.b',wpts(1,:),wpts(2,:),'or')
xlabel('X')
ylabel('Y')
```







به پیوست، فایل های متلب مربوط به این تمرین خدمتتان ارسال می‌شود.

ممنون از توجه شما