باسمه تعالی

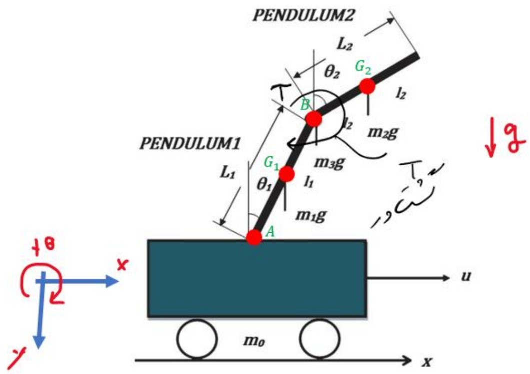
علی مجیبی ۴۰۱۲۰۲۶۱۸

تمرین پاندول معکوس دوگانه درس
کنترل پیشرفته

---

۱. استخراج معادلات دینامیکی سیستم

از روش لاگرانژ استفاده می‌کنیم و معادلات حاکم بر سیستم را به فرم $\mathbf{M\ddot{q}} + \mathbf{Bias} = \mathbf{Q}$ می‌نویسیم.



سرعت مرکز جرم پاندول ۱ و ۲ با استفاده از رابطه (۱) بدست می‌آید.

$$\mathbf{v}_{G1} = \begin{pmatrix} \dot{x} + \dfrac{L\,\dot{\theta}_1\,\cos(\theta_1)}{2} \\ \dfrac{L\,\dot{\theta}_1\,\sin(\theta_1)}{2} \\ 0 \end{pmatrix}, \mathbf{v}_{G2} = \begin{pmatrix} \dot{x} + L\,\dot{\theta}_1\,\cos(\theta_1) + \dfrac{L\,\dot{\theta}_2\,\cos(\theta_2)}{2} \\ L\,\dot{\theta}_1\,\sin(\theta_1) + \dfrac{L\,\dot{\theta}_2\,\sin(\theta_2)}{2} \\ 0 \end{pmatrix} \quad (۱)$$

انرژی جنبشی سیستم

$$T = \frac{1}{2}m\mathbf{v}_{G1}^T\mathbf{v}_{G1} + \frac{1}{2}I_1\dot{\theta}_1^{\,2} + \frac{1}{2}m\mathbf{v}_{G2}^T\mathbf{v}_{G2} + \frac{1}{2}I_2\dot{\theta}_2^{\,2} + \frac{1}{2}m_0\dot{x}^2 \quad (۲)$$

انرژی پتانسیل سیستم

$$V = \ g\,m\left(L\cos(\theta_1) + \frac{L\cos(\theta_2)}{2}\right) + \frac{L\,g\,m\cos(\theta_1)}{2} \quad (۳)$$

لاگرانژین

$$L = T - V \quad (۴)$$

رابطه لاگرانژ

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) + \frac{\partial L}{\partial q} = Q_i^{nc} \quad (۵)$$

فرآیند بالا با دستورات سمبولیک در متلب کدنویسی شده است (فایل symbolic_derivation.m)، در نهایت معادلات دینامیکی سیستم به فرم ماتریسی رابطه (۶) بدست می‌آیند.

$$\begin{pmatrix} 2\,m+m_0 & \frac{3\,L\,m\,\cos(\theta_1)}{2} & \frac{L\,m\,\cos(\theta_2)}{2} \\ \frac{3\,L\,m\,\cos(\theta_1)}{2} & \frac{4\,L^2\,m}{3} & \frac{mL^2\cos(\theta_1-\theta_2)}{2} \\ \frac{L\,m\,\cos(\theta_2)}{2} & \frac{mL^2\cos(\theta_1-\theta_2)}{2} & \frac{L^2\,m}{3} \end{pmatrix}\begin{pmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} + \begin{pmatrix} -\frac{L\,m\left(3\,\sin(\theta_1)\,\dot{\theta}_1^{\,2}+\sin(\theta_2)\,\dot{\theta}_2^{\,2}\right)}{2} \\ -\frac{L\,m\left(3\,g\,\sin(\theta_1)-L\,\dot{\theta}_2^{\,2}\,\sin(\theta_1-\theta_2)\right)}{2} \\ -\frac{L\,m\left(L\,\sin(\theta_1-\theta_2)\,\dot{\theta}_1^{\,2}+g\,\sin(\theta_2)\right)}{2} \end{pmatrix} = \begin{pmatrix} u \\ \tau \\ -\tau \end{pmatrix} \qquad (۶)$$

فرم فضای حالت غیرخطی معادله (۶) را می‌توان با بدست آورد $\begin{pmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix}$ به‌صورت صریح از رابطه بالا بدست آورد، یعنی:

$$\ddot{\mathbf{q}} = \begin{pmatrix} \ddot{x} \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{pmatrix} = \mathbf{M}^{-1}(\mathbf{Q}-\mathbf{Bias}) \qquad (۷)$$

متغیرهای حالت را به‌صورت زیر درنظر می‌گیریم:

$$\mathbf{x} = [x_1, x_2, x_3, x_4, x_6]^T = \left[x, \dot{x}, \theta_1, \dot{\theta}_1, \dot{\theta}_2\right]^T \qquad (۸)$$

برای بدست آوردن فرم فضای حالت غیرخطی به‌صورت زیر کدنویسی انجام شده است:

```
% --------------------------State space Modeling---------------------------
X = [x x_dot theta_1 theta_1_dot theta_2 theta_2_dot]% state variables
X_dot = [x_dot, ddq(1), theta_1_dot, ddq(2), theta_2_dot, ddq(3)].'; % deriviative of state variables
u_vec = [u tau].';% inputs vectors
B = simplify(jacobian(X_dot, u_vec))
A = simplify(jacobian(X_dot, X))
```

که درنهایت فرم غیرخطی ماتریس‌هایA و B را به ما می‌دهد. از آنجایی که فرم غیرخطی این ماتریس‌های بسیار طولانی است از آوردن آن در متن گزارش خودداری می‌شود.

۲. خطی سازی معادلات حاکم

با داشتن فرم غیرخطی ماتریس A و B می‌توان به‌صورت زیر فرم خطی شده آن را بدست آورد:

```
A_lin = subs(A, [x x_dot theta_1 theta_1_dot theta_2 theta_2_dot], zeros(1, 6))
B_lin = subs(B, [x x_dot theta_1 theta_1_dot theta_2 theta_2_dot], zeros(1, 6))
```

به عبارتی در همان فرم غیرخطی که برای ماتریس‌ها بدست آمد، به‌جای متغیرهای حالت صفر قرار می‌دهیم، در نهایت این ماتریس‌ها برای سیستم حطی سازی شده به فرم زیر بدست می‌آیند.

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\dfrac{54\,g\,m}{8m+28m_0} & 0 & \dfrac{6\,g\,m}{8m+28m_0} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{6\,(15\,L\,g\,m^2+12\,L\,g\,m_0\,m)}{mL^2(8m+28m_0)} & 0 & -\dfrac{6\,(3\,L\,g\,m^2+6\,L\,g\,m_0\,m)}{mL^2(8m+28m_0)} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\dfrac{6\,(9\,L\,g\,m^2+18\,L\,g\,m_0\,m)}{mL^2(8m+28m_0)} & 0 & \dfrac{6\,(5\,L\,g\,m^2+16\,L\,g\,m_0\,m)}{mL^2(8m+28m_0)} & 0 \end{pmatrix} \tag{۹}$$

$$\mathbf{B} = \begin{pmatrix} 0 & 0 \\ \dfrac{28}{8\,m+28\,m_0} & -\dfrac{48}{L(8m+28m_0)} \\ 0 & 0 \\ -\dfrac{36}{L(8m+28m_0)} & \dfrac{6\,(16\,m+20\,m_0)}{L^2\,m\,(8\,m+28\,m_0)} \\ 0 & 0 \\ \dfrac{12}{L(8m+28m_0)} & -\dfrac{6\,(16\,m+44\,m_0)}{L^2\,m\,(8\,m+28\,m_0)} \end{pmatrix} \tag{۱۰}$$

فضای حالت خطی را می‌توان به‌صورت زیر نوشت

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ y &= \mathbf{Cx} \end{aligned} \tag{۱۱}$$

با جایگذاری مقادیر عددی داده شده داریم:

```
A =

         0    1.0000         0         0         0         0
         0         0   -4.4145         0    0.4905         0
         0         0         0    1.0000         0         0
         0         0   61.8030         0  -26.4870         0
         0         0         0         0         0    1.0000
         0         0  -79.4610         0   67.6890         0
```

```
B =

         0         0
    0.4667   -1.6000
         0         0
   -1.2000   38.4000
         0         0
    0.4000  -76.8000
```

۳. بررسی کنترل پذیری

برای بررسی کنترل‌پذیری به‌صورت زیر عمل می‌کنیم:

```
% n - m = 4
M_c = [B, A*B, A^2*B, A^3*B, A^4*B]% Controlability matrix
rank(M_c) % M_c is full rank ⟶ OK
```

که ماتریس کنترل پذیری به‌صورت زیر بدست می‌آید و رتبه کامل است.

```
M_c =

   1.0e+05 *

         0         0    0.0000   -0.0000         0         0    0.0001   -0.0021         0         0
    0.0000   -0.0000         0         0    0.0001   -0.0021         0         0    0.0043   -0.2350
         0         0   -0.0000    0.0004         0         0   -0.0008    0.0441         0         0
   -0.0000    0.0004         0         0   -0.0008    0.0441         0         0   -0.0848    4.9091
         0         0    0.0000   -0.0008         0         0    0.0012   -0.0825         0         0
    0.0000   -0.0008         0         0    0.0012   -0.0825         0         0    0.1502   -9.0864
```

۴. مشاهده پذیری سیستم با فرض $y = [x, \theta_1]^T$

برای این قسمت به‌صورت زیر عمل میکنیم:

```
% ---------------------------- part 04 ----------------------------------
% y = [x theta_1]
C_1 = [1 0 0 0 0 0;0 0 1 0 0 0];
N_1 = [C_1; C_1*A; C_1*A^2; C_1*A^3; C_1*A^4]
rank(N_1) % full rank ⟶ Observable
C = C_1;
```

ماتریس مشاهده‌پذیری به‌صورت زیر و رتبه کامل است.

```
>> N_1

N_1 =

   1.0e+03 *

    0.0010         0         0         0         0         0
         0         0    0.0010         0         0         0
         0    0.0010         0         0         0         0
         0         0         0    0.0010         0         0
         0         0   -0.0044         0    0.0005         0
         0         0    0.0618         0   -0.0265         0
         0         0         0   -0.0044         0    0.0005
         0         0         0    0.0618         0   -0.0265
         0         0   -0.3118         0    0.1501         0
         0         0    5.9243         0   -3.4299         0
```

۵. مشاهده پذیری سیستم با فرض $y = [\theta_1, \theta_2]^T$

برای این قسمت به‌صورت زیر عمل می‌شود:

```
% ---------------------------- part 05 ----------------------------------
% y = [theta_1 theta_2]
C_2 = [0 0 1 0 0 0;0 0 0 0 1 0];
N_2 = [C_2; C_2*A; C_2*A^2; C_2*A^3; C_2*A^4]
rank(N_2) %rank=4 --> Not Observable
```

ماتریس مشاهده پذیری در این حالت رتبه کامل نیست و سیستم مشاهده پذیر نخواهد بود.

۶. طراحی رگولاتور

روش EESA به‌صورت زیردرمتلب پیاده سازی شده است.

```
% ---------------------------- part 06 ----------------------------------
% EESA
mu(1) = -2; mu(2) = -2; mu(3) = -2-j; mu(4) = -2+j; mu(5) = -3; mu(6) = -3;
S1 = null([A-mu(1)*eye(6) B])
S3 = null([A-mu(3)*eye(6) B])
S4 = null([A-mu(4)*eye(6) B])
S5 = null([A-mu(5)*eye(6) B])
v1 = S1(1:6, 1); q1 = S1(7:end, 1);
v2 = S1(1:6, 2); q2 = S1(7:end, 2);
v3 = S3(1:6, 1); q3 = S3(7:end, 1);
v4 = S4(1:6, 1); q4 = S4(7:end, 1);
v5 = S5(1:6, 1); q5 = S5(7:end, 1);
v6 = S5(1:6, 2); q6 = S5(7:end, 2);

K_EESA = -[q1 q2 q3 q4 q5 q6]*inv([v1 v2 v3 v4 v5 v6])
K_EESA=real(K_EESA);
eig(A-B*K_EESA)
```

بهره کنترلی بدست آمده در این حالت به‌صورت زیر است:

```
>> K_EESA

K_EESA =

   -4.5199   -7.3714  -57.7888  -12.7990  -20.5140   -5.5092
   -0.0581   -0.0940    0.5112   -0.1250   -1.1424   -0.1163
```

۷. طراحی مشاهده‌گر

الگوریتم GCCF به‌صورت زیر پیاده سازی شده است.

```
% ---------------------------- part 07 -----------------------------------
% Observer Design using GCCf algorithm
A = A.'; B=C_1.';
mu = -10*ones(1, 6);
M_h = [B(:, 1) A*B(:, 1) A^2*B(:, 1) A^3*B(:, 1), B(:, 2) A*B(:, 2)] % gamma1 = 4, gamma2 = 2
M_hi = inv(M_h);
e14T=M_hi(4, :)
e22T=M_hi(4+2, :)
Ti = [e14T; e14T*A; e14T*A^2; e14T*A^3; e22T; e22T*A]
T=inv(Ti)
% dX = AX + Bu ⟶ dZ = A_G Z + B_G v; X=TZ u=fw w = H - vZ
Ag1 = [zeros(3, 1), eye(3);zeros(1, 4)];
Ag2 = [0 1;0 0];
A_G = [Ag1, zeros(4, 2);zeros(2, 4) Ag2]
Bg1 = [0 0 0 1].';
Bg2 = [0 1].';
B_G = [Bg1 zeros(4,1);zeros(2,1) Bg2];
% Ti*B*F = B_G ⟶ B*F = T*B_G⟶(B.'*B)*F=B.'*T*B_G⟶F = inv(B.'*B)*B.'*T*B_G
F = inv(B.'*B)*B.'*T*B_G;
H = B_G.'*(A_G-Ti*A*T);

%desired polynomials
syms s
P1 = flip(coeffs(expand((s-mu(1))^4)));% gamma1 = 4
P2 = flip(coeffs(expand((s-mu(1))^2)));% gamma2 = 2
Ad1 = [zeros(3,1), eye(3);-flip(P1(2:end))]
Ad2 = [0 1;-flip(P2(2:end))]
A_d =[Ad1, zeros(4,2);zeros(2, 4), Ad2]

Gamma = B_G.'*(A_G-A_d)
K_o = double(F*(Gamma-H)*Ti).'
```

بهره کنترلی بدست آمده در این حالت به‌صورت زیر است:

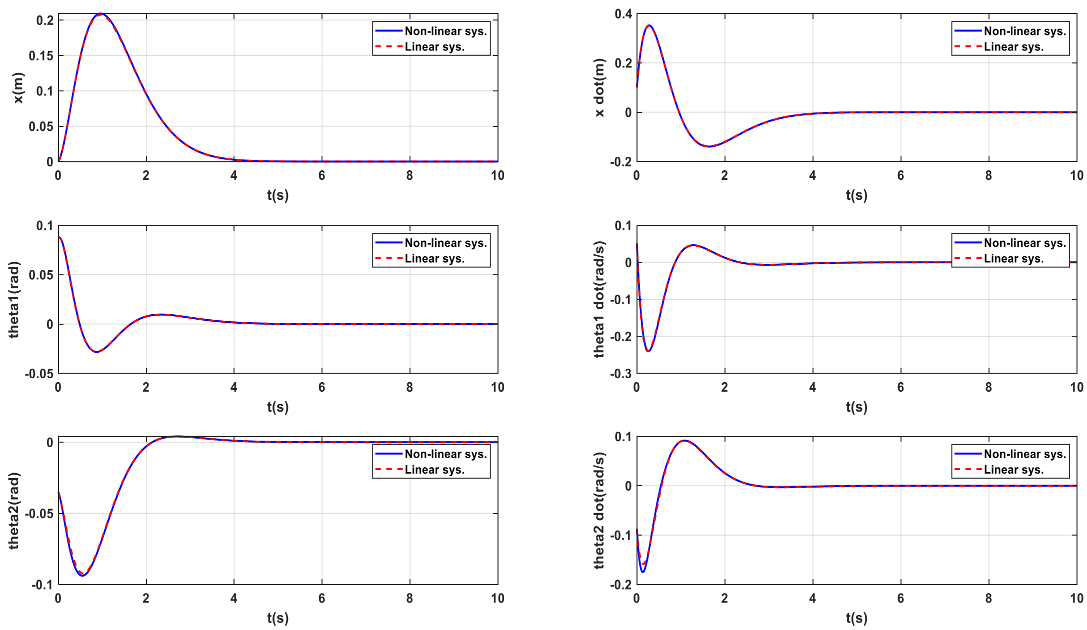
```
>> K_o

K_o =

   1.0e+05 *

    0.0004         0
    0.0091         0
   -0.0108    0.0002
   -0.4353   -0.0008
    0.2339    0.0018
    1.9403    0.0143
```

۸. پاسخ سیستم خطی و غیر خطی همراه با رگولاتور به شرایط اولیه

در این حالت فرمان کنترلی به‌صورت رابطه (۱۹) خواهد بود

$$u = -\mathbf{Kx} \tag{۱۲}$$

با اعمال گین کنترلی بدست آمده به پاسخ‌های زیر برای سیستم‌های خطی و غیرخطی می‌رسیم.



شکل ۱ پاسخ سیستم های خطی و غیرخطی به شرایط اولیه درحضور رگولاتور طراحی شده

۹. استفاده از مشاهده‌گر برای اعمال کنترلر به سیستم

در این حالت فرمان کنترلی به‌صورت رابطه (۲۰) خواهد بود

$$u = -\mathbf{K}\hat{\mathbf{x}} \tag{۱۳}$$

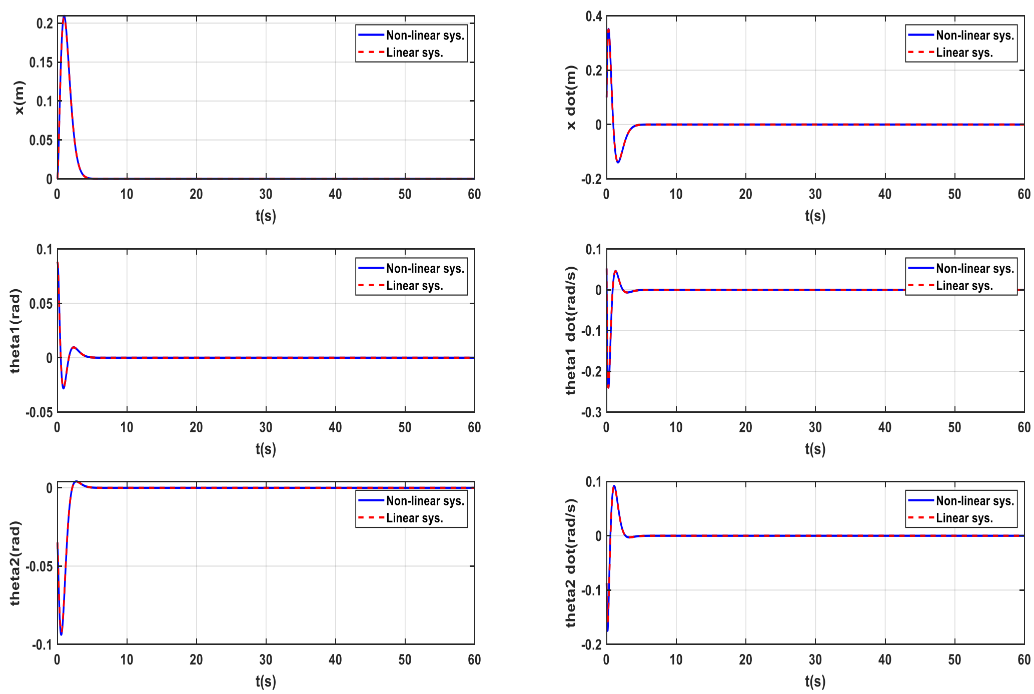
در این رابطه $\hat{\mathbf{x}}$ تخمین متغیرهای حالت است. با استفاده از مشاهده‌گر مرتبه کامل لومبرگر برای سیستم خطی می‌توان نوشت:

$$\hat{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}(-\mathbf{K}\hat{\mathbf{x}}) + \mathbf{K_o}\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \tag{۱۴}$$

و برای سیستم غیرخطی:

$$\dot{\hat{\mathbf{x}}} = \mathbf{f}(\hat{\mathbf{x}}) + \mathbf{B}(\hat{\mathbf{x}}) \times (-\mathbf{K}\hat{\mathbf{x}}) + \mathbf{K_o}\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \tag{۱۵}$$

با حل دستگاه معادلات (۲۱) و (۱۳) برای سیستم خطی به نتایج نشان داده شده در شکل زیر می‌رسیم. توجه شود که برای پایدار شدن سیستم غیرخطی شرایط اولیه مشاهده‌گر و سیستم یکی در نظر گرفته شده است.



شکل ۲ پاسخ سیستم خطی و غیرخطی به شرایط اولیه در حضور رگولاتور همراه با مشاهده‌گر

۱۰.طراحی سروو کنترل نوع ۱ با استفاده از پیشخوراند همراه با مشاهده‌گر

طراحی کنترلر با استفاده از روش GCCF به‌صورت زیر انجام شده است:

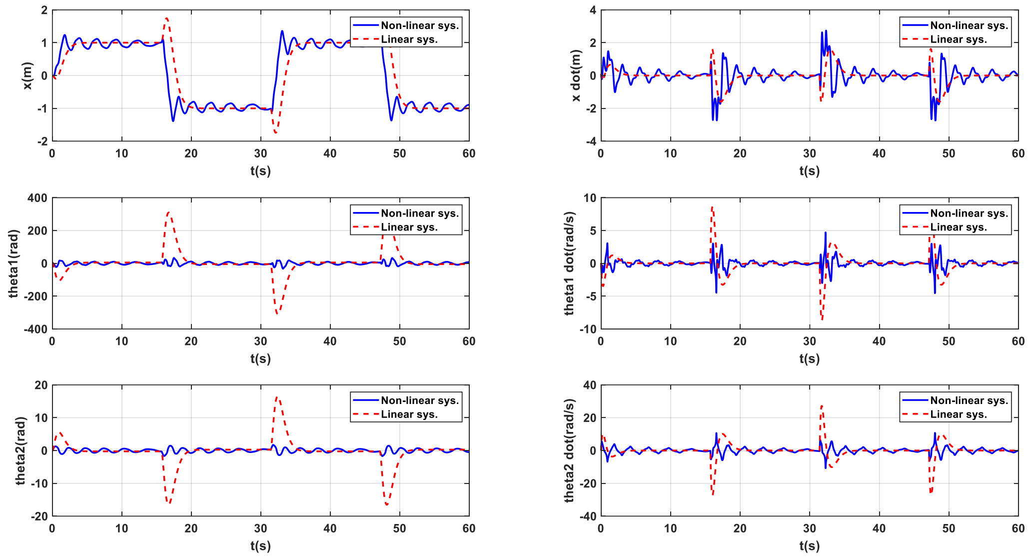
```
% ---------------------------- part 10 ----------------------------------
% Regulator Design using GCCf algorithm
mu(1) = -2; mu(2) = -2; mu(3) = -2-j; mu(4) = -2+j; mu(5) = -3; mu(6) = -3;
M_h = [B(:, 1) A*B(:, 1) A^2*B(:, 1) A^3*B(:, 1), B(:, 2) A*B(:, 2)] % gamma1 = 4, gamma2 = 2
M_hi = inv(M_h);
e14T=M_hi(4, :)
e22T=M_hi(4+2, :)
Ti = [e14T; e14T*A; e14T*A^2; e14T*A^3; e22T; e22T*A]
T=inv(Ti)
% dX = AX + Bu ⟶ dZ = A_G Z + B_G v; X=TZ u=fw w = H - vZ
Ag1 = [zeros(3, 1), eye(3);zeros(1, 4)];
Ag2 = [0 1;0 0];
A_G = [Ag1, zeros(4, 2);zeros(2, 4) Ag2]
Bg1 = [0 0 0 1].';
Bg2 = [0 1].';
B_G = [Bg1 zeros(4,1);zeros(2,1) Bg2];
% Ti*B*F = B_G ⟶ B*F = T*B_G⟶(B.'*B)*F=B.'*T*B_G⟶F = inv(B.'*B)*B.'*T*B_G
F = inv(B.'*B)*B.'*T*B_G;
H = B_G.'*(A_G-Ti*A*T);

%desired polynomials
syms s
P1 = flip(coeffs(expand((s-mu(1))*(s-mu(2))*(s-mu(3))*(s-mu(4)))));% gamma1 = 4
P2 = flip(coeffs(expand((s-mu(5))*(s-mu(6)))));% gamma2 = 2
Ad1 = [zeros(3,1), eye(3);-flip(P1(2:end))]
Ad2 = [0 1;-flip(P2(2:end))]
A_d =[Ad1, zeros(4,2);zeros(2, 4), Ad2]

Gamma = B_G.'*(A_G-A_d)
K_GCCF = double(F*(Gamma-H)*Ti)
```

$$u = -\mathbf{K}\hat{\mathbf{x}} + u_{ff} \tag{۱۶}$$

که در آن

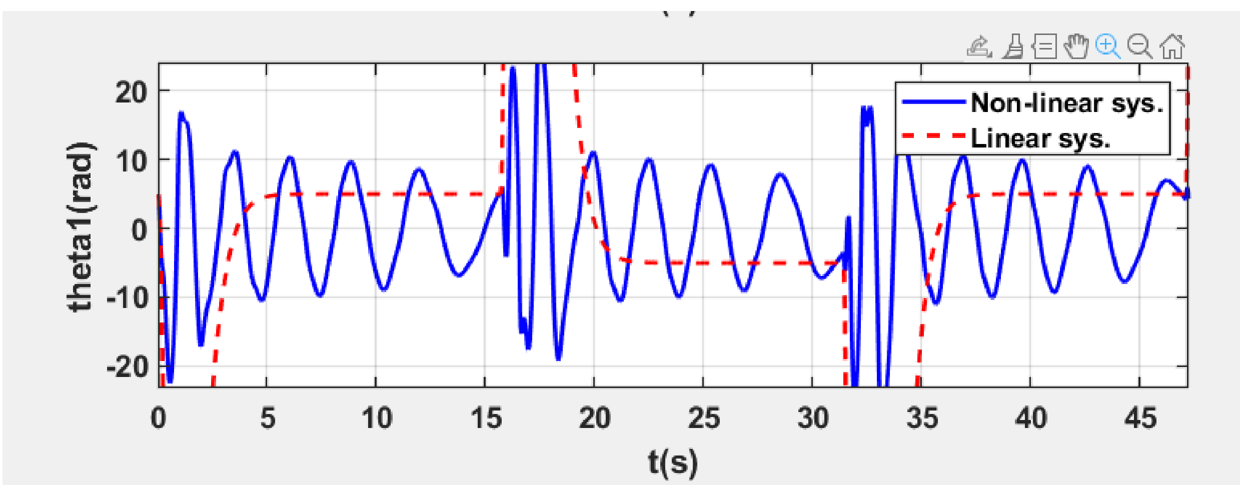
$$u_{ff} = -\mathbf{C}(\mathbf{A} - \mathbf{BK})^{-1}\mathbf{Br} \tag{۱۷}$$

پاسخ سیستم‌های خطی و غیرخطی در شکل زیر نشان داده شده است. مشخص است که سیگنال رفرنس برای متغیر حالت x برای هردو سیستم خطی و غیرخطی به خوبی دنبال می‌شود. تنها مسأله فراجهش‌های زیاد سیستم خطی سازی شده است که غیرواقعی به‌نظر می‌رسد.



شکل ۳ عملکرد سروو کنترلر نوع ۱ با پیشخوراند برای سیستم های خی و غیرخطی

دقت شود که متغیر θ_1 هم برای سیستم خطی مقدار خواسته شده را تعقیب می‌کند اما به علت فراجهش زیاد در شکل مشخص نشده است، برای نمایش بهتر به شکل زیر توجه کنید. در اینجا سیستم غیرخطی حول مقادیر خواسته شده نوسان می‌کند. همچنین برای پایدار شدن سیستم غیرخطی شرایط اولیه مشاهده‌گر و سیستم اصلی مشابه هم درنظر گرفته شده است.



شکل ۴ تغییرات متغیر θ_1

۱۱.طراحی سروو کنترل نوع ۱ با استفاده از انتگرال‌گیر همراه با مشاهده‌گر

معادلات حالت سیستم خطی همراه با انتگرال‌گیر را می‌توان به‌صورت رابطه (۲۷) نوشت:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u \\ \boldsymbol{\xi} &= \mathbf{r} - \mathbf{C}\mathbf{x} \\ \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}u + \mathbf{K}_o\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \\ u &= -\mathbf{K}\begin{bmatrix} \hat{\mathbf{x}} \\ \boldsymbol{\xi} \end{bmatrix} \end{aligned} \tag{۱۸}$$

به‌صورت مشابه برای سیستم غیرخطی خواهیم داشت:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{B}(\hat{\mathbf{x}})u \\ \boldsymbol{\xi} &= \mathbf{r} - \mathbf{C}\mathbf{x} \\ \dot{\hat{\mathbf{x}}} &= \mathbf{f}(\hat{\mathbf{x}}) + \mathbf{B}(\hat{\mathbf{x}})u + \mathbf{K}_o\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \\ u &= -\mathbf{K}\begin{bmatrix} \hat{\mathbf{x}} \\ \boldsymbol{\xi} \end{bmatrix} \end{aligned} \tag{۱۹}$$

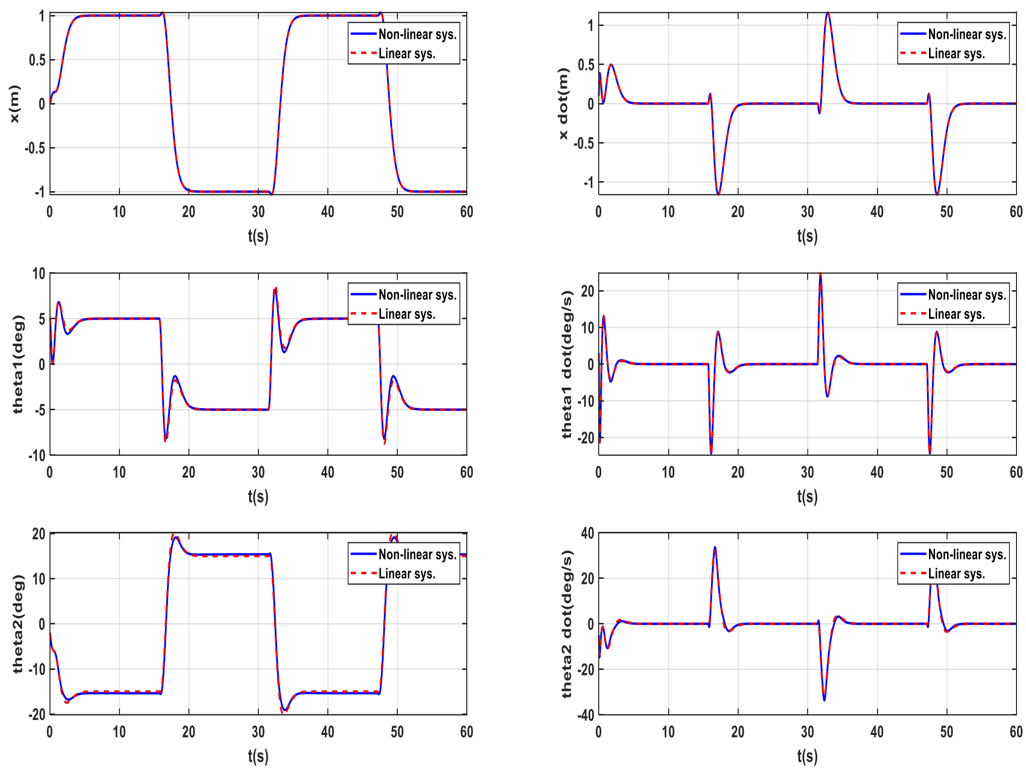
در این روابط بهره کنترلی $\mathbf{K}$ با استفاده از الگوریتم EESA برای زوج ماتریس رابطه زیر محاسبه می‌شود.

$$\tilde{A} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix}, \tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \tag{۲۰}$$

الگوریتم *EESA* به صورت زیر در متلب پیاده سازی شده است:

```
% ----------------------------- part 11 -----------------------------------
% Adding Integrator to the system, using EESA algorithm for pole placement
A_new = [A_lin(L,g,m,m_0), zeros(6, 2); -C, zeros(2,2)];
B_new = [B_lin(L,m,m_0);zeros(2, 2)];
C = [1 0 0 0 0 0;0 0 1 0 0 0];

mu(1) = -2; mu(2) = -2; mu(3) = -2-j; mu(4) = -2+j; mu(5) = -3; mu(6) = -3; mu(7)=-4; mu(8)=-4;
S1 = null([A_new-mu(1)*eye(8) B_new])
S3 = null([A_new-mu(3)*eye(8) B_new])
S4 = null([A_new-mu(4)*eye(8) B_new])
S5 = null([A_new-mu(5)*eye(8) B_new])
S7 = null([A_new-mu(7)*eye(8) B_new])
v1 = S1(1:8, 1); q1 = S1(9:end, 1);
v2 = S1(1:8, 2); q2 = S1(9:end, 2);
v3 = S3(1:8, 1); q3 = S3(9:end, 1);
v4 = S4(1:8, 1); q4 = S4(9:end, 1);
v5 = S5(1:8, 1); q5 = S5(9:end, 1);
v6 = S5(1:8, 2); q6 = S5(9:end, 2);
v7 = S7(1:8, 1); q7 = S7(9:end, 1);
v8 = S7(1:8, 2); q8 = S7(9:end, 2);
%VV = [v1 v2 v3 v4 v5 v6 v7 v8]
%rank(VV)

K_EESA = -[q1 q2 q3 q4 q5 q6 q7 q8]*inv([v1 v2 v3 v4 v5 v6 v7 v8]);
K_EESA=real(K_EESA);
eig(A_new-B_new*K_EESA); % OK
```

با حل معادلات دیفراسیل برای سیستم‌های خطی و غیرخطی پاسخ به‌صورت نشان داده شده در شکل زیر خواهد بود. توجه شود که در این قسمت هم برای پایدار شدن سیستم غیرخطی شرایط اولیه مشاهده‌گر و سیستم اصلی مشابه هم درنظر گرفته شده است. دقت شود که در این قسمت متغیر θ_2 علی‌رغم خواست ما به صفر نمی‌رود بلکه به‌نوعی دارد یک ورودی ناخواسته را تعقیب می‌کند.



شکل ۵ طراحی کنترلر سروو انتگرال گیر همراه با مشاهده گر