به نام خدا

دانشگاه صنعتی خواجه نصیرالدین طوسی

دانشکده برق

کنترل مدرن

گزارش پروژه نهایی

سیده زهرا عربی

۴۰۰۰۷۱۷۳

استاد : آقای دکتر حمید رضا تقی راد

بهمن ۱۴۰۲

https://github.com/Zahra-Arabi/Examining-the-equations-of-state-of-the-Segway-system.git

فهرست مطالب

چکیده

$Segway$ یک وسیله نقلیه دوچرخ و خودمتعادل‌کننده است که توسط Dean Kamen طراحی و توسط شرکت .Segway Inc ساخته شده است. این وسیله نقلیه الکتریکی از سیستم‌های ژیروسکوپ و حسگرهای پیشرفته برای حفظ تعادل خود استفاده می‌کند و با حرکت دادن دسته به جلو یا عقب حرکت می‌کند. از $Segway$ به عنوان وسیله نقلیه‌ای برای حمل و نقل شخصی در محیط‌های شهری، گردشگری، و برخی کاربردهای تجاری مانند گشت‌های امنیتی استفاده می‌شود. مزایای اصلی آن شامل کاهش آلودگی، صرفه‌جویی در زمان و انرژی، و سهولت استفاده است.

سوال اول

مدل کامل سیستم $Segway$ را به دست آورید، فرم فضای حالت را تشکیل داده و ورودی(ها) و خروجی(ها)ی مناسب برای آن را تعیین کنید. هم چنین اگر می توانید، سیستم را در قالب یک بلوک دیاگرام نمایش دهید .در ادامه، سیستم انتخابی خود را حول یک نقطهٔ کار مناسب خطی سازی کنید و معادلات را به فرم متعارف فضای حالت تبدیل کنید. هم چنین اگر می توانید، محدودهٔ معتبر سیستم خطی را هم به صورت نظری مشخص کنید.

معادلات دینامیکی سیستم Segway

فرض می‌کنیم که $Segway$ شامل یک چرخ و بدنه است که با زاویه θ نسبت به حالت عمودی قرار دارد و x مکان افقی چرخ است.

معادلات دینامیکی را می‌توان با استفاده از قوانین نیوتن و لاگرانژ به دست آورد. متغیرهای زیر را تعریف می‌کنیم:

- M: جرم بدنه
- m: جرم چرخ
- l: فاصله مرکز جرم بدنه تا محور چرخ
- I: ممان اینرسی بدنه حول مرکز جرم
- θ: زاویهٔ انحراف بدنه از حالت عمودی
- x: موقعیت افقی چرخ
- u: نیروی اعمالی توسط موتور

با توجه به این فرضیات، معادلات دینامیکی سیستم به صورت زیر می‌باشند:

$$u = ml\theta'^2 sin(\theta) - ml\theta'' cos(\theta) + (M + m)x''$$

$$mglsin(\theta) = ml\theta'' cos(\theta) + I\theta''$$

فرم فضای حالت

برای تشکیل فرم فضای حالت، متغیرهای حالت زیر را تعریف می‌کنیم:

(موقعیت افقی) $x = x_1$

(سرعت افقی) $x' = x_2$

(زا.یه انحراف) $\theta = x_3$

(سرعت زاویه ای)) $\theta' = x_4$

معادلات فضای حالت را می‌توان به صورت زیر نوشت:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{\left(\frac{mgl\, sin\theta}{I^2+ml}\right) u - ml\, sin\theta\, x_4^2 + ml\, cos\theta}{\frac{m^2 l^2 cos^2\theta}{I^2+ml} - M + m}$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{\left(\frac{u_4^2 - ml\, sin\theta\, x}{M+m}\right) mgl\, sin\theta - ml\, cos\theta}{\frac{ml^2 cos^2\theta}{M+m} - I^2 + ml}$$

خطی سازی حول نقطه تعادل

برای خطی سازی سیستم، فرض می‌کنیم که θ و θ′ کوچک هستند. بنابراین، از تقریب‌های $sin\theta \approx \theta$ و $cos\theta \approx 1$ استفاده می‌کنیم.

در نتیجه معادلات خطی شده فضای حالت به صورت زیر خواهند بود:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{u + ml\theta}{M + m}$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{\left(\frac{u}{M+m}\right) g\theta - ml(M + m)}{I^2 + ml}$$

این معادلات را می‌توان به فرم استاندارد فضای حالت نوشت:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{ml}{M+m} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g(M+m)}{I^2+ml} & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{M+m} \\ 0 \\ \frac{ml}{(I^2+ml)(M+m)} \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} x$$

محدوده معتبر سیستم خطی

محدوده معتبر خطی سازی حول نقطه تعادل کوچک بودن θ و θ' است. بنابراین، در کاربردهای واقعی، این تقریب در محدوده $|\theta| \ll 1$ و $|\theta'| \ll 1$ معتبر خواهد بود. به عبارت دیگر، این خطی سازی برای زوایای کوچک و تغییرات سریع معتبر است.

سوال دوم

ماتریس انتقال حالت سیستم $Segway$ را به دست آورید .پاسخ حلقه باز سیستم خطی سازی شدهٔ خود را به ازای حداقل دو ورودی مختلف و شرایط اولیهٔ متفاوت بررسی و تحلیل کنید .در ادامه، سیستم غیرخطی خود را وارد محیط سیمولینک متلب کنید و پاسخ حلقه باز سیستم را به ازای حداقل دو ورودی مختلف بررسی کنید. همین کار را با قراردادنِ ماتریس سیستم خطی شده در متلب تکرار کنید و نتایج خود را راستی آزمایی، تحلیل و مقایسه کنید .اگر می توانید، محدودهٔ معتبربودن سیستم خطی سازی شدهٔ خود را بیابید .

محاسبه ماتریس انتقال حالت

برای محاسبه ماتریس انتقال حالت، از معادلات فضای حالت خطی شده استفاده می‌کنیم. ماتریس انتقال حالت برای یک سیستم خطی با ماتریس A به صورت e^{At} تعریف می‌شود.

کد متلب برای محاسبه ماتریس انتقال حالت

در متلب، از دستور $expm$ برای محاسبه ماتریس انتقال حالت استفاده می‌شود.

```
% تعریف پارامترهای سیستم
M = 1;   % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0; 0, 0, 1, 0];

D = [0; 0];


% محاسبه ماتریس انتقال حالت برای زمان  t
syms t;
Phi_t = expm(A * t);  % محاسبه ماتریس انتقال حالت برای هر زمان
disp(Phi_t);
```

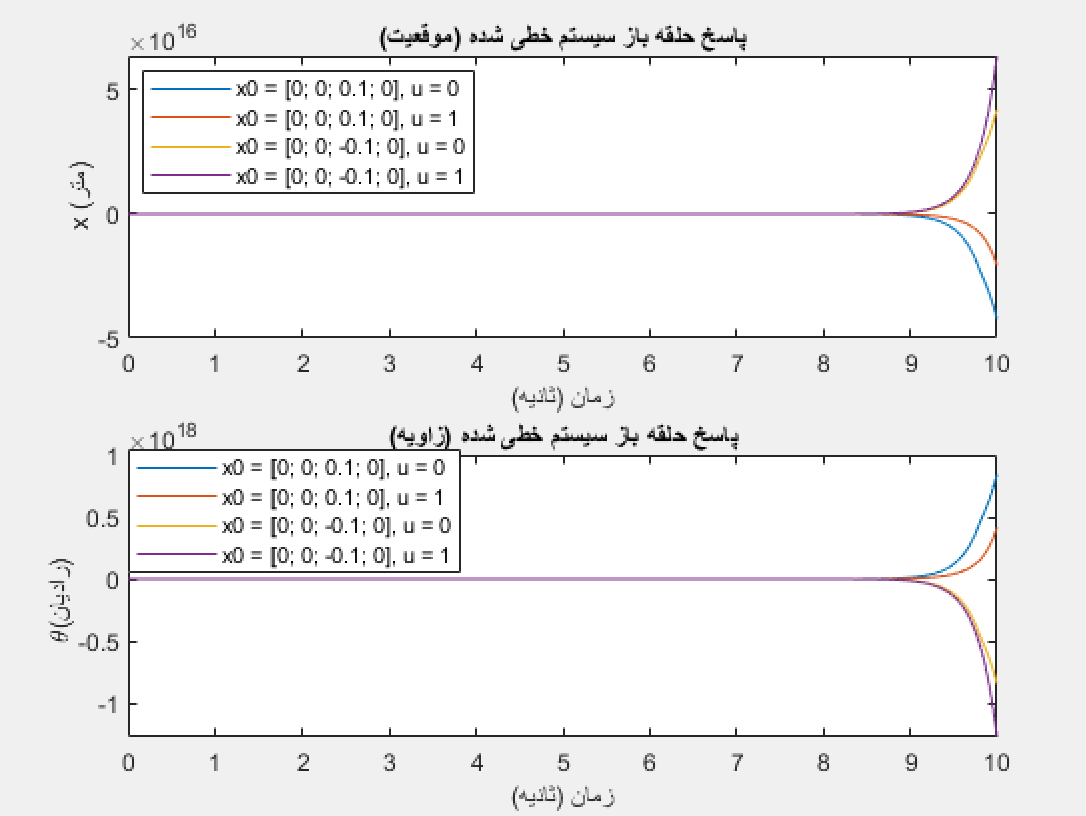
$$e^{At} = \begin{pmatrix} 1 & t & 0.0001305\cosh(18.65\,t) - 0.0001305 & -0.0001305\,t + 6.998\times 10^{-6}\sinh(18.65\,t) \\ 0 & 1 & 0.00244\sinh(18.65\,t) & 0.0001305\cosh(18.65\,t) - 0.0001305 \\ 0 & 0 & \cosh(18.65\,t) & 0.05359\sinh(18.65\,t) \\ 0 & 0 & 18.65\sinh(18.65\,t) & \cosh(18.65\,t) \end{pmatrix}$$

پاسخ حلقه باز سیستم خطی شده

برای بررسی پاسخ حلقه باز، باید معادلات حالت را با شرایط اولیه مختلف و ورودی‌های متفاوت شبیه‌سازی کنیم.

کد متلب برای شبیه‌سازی پاسخ حلقه باز

```
% پارامترهای سیستم خطی شده
A = [0 1 0 0; 0 0 -0.98 0; 0 0 0 1; 0 0 19.6 0];   % ماتریس A
B = [0; 1; 0; -1];   % ماتریس B

% شرایط اولیه
x0_1 = [0; 0; 0.1; 0];   % شرایط اولیه اول
x0_2 = [0; 0; -0.1; 0];   % شرایط اولیه دوم

% ورودی‌ها
u1 = 0;   % ورودی اول (بدون ورودی)
u2 = 1;   % ورودی دوم (ورودی ثابت)

% بازه زمانی شبیه‌سازی
t = [0 10];   % از ۰ تا ۱۰ ثانیه

% شبیه‌سازی سیستم خطی شده
[T, X1] = ode45(@(t, x) A*x + B*u1, t, x0_1);
[T, X2] = ode45(@(t, x) A*x + B*u2, t, x0_1);
[T, X3] = ode45(@(t, x) A*x + B*u1, t, x0_2);
[T, X4] = ode45(@(t, x) A*x + B*u2, t, x0_2);

% نمایش نتایج
figure;
subplot(2,1,1);
plot(T, X1(:, 1), T, X2(:, 1), T, X3(:, 1), T, X4(:, 1));
title('پاسخ حلقه باز سیستم خطی شده (موقعیت)');
xlabel('زمان (ثانیه)');
ylabel('x (متر)');
legend('x0 = [0; 0; 0.1; 0], u = 0', 'x0 = [0; 0; 0.1; 0], u = 1', 'x0 = [0; 0; -0.1; 0], u = 0', 'x0 = [0; 0; -0.1; 0], u = 1');

subplot(2,1,2);
plot(T, X1(:, 3), T, X2(:, 3), T, X3(:, 3), T, X4(:, 3));
title('پاسخ حلقه باز سیستم خطی شده (زاویه)');
xlabel('زمان (ثانیه)');
ylabel('\theta (رادیان)');
legend('x0 = [0; 0; 0.1; 0], u = 0', 'x0 = [0; 0; 0.1; 0], u = 1', 'x0 = [0; 0; -0.1; 0], u = 0', 'x0 = [0; 0; -0.1; 0], u = 1');
```



نمودارها

نمودارها پاسخ حلقه باز سیستم را به دو ورودی مختلف و دو شرط اولیه مختلف نشان می‌دهند.

پاسخ موقعیت

نمودار اول پاسخ موقعیت سیستم را نشان می‌دهد .این نمودار نشان می‌دهد که:

- اگر هیچ ورودی به سیستم اعمال نشود(u1 = 0) ، پاسخ سیستم به شرایط اولیه بستگی دارد . اگر حالت دوم در ابتدا مثبت باشد(x0_1 = [0; 0; 0.1; 0]) ، موقعیت سیستم به طور نامحدود افزایش می‌یابد .اگر حالت دوم در ابتدا منفی باشد(x0_2 = [0; 0; -0.1; 0]) ، موقعیت سیستم به طور نامحدود کاهش می‌یابد.
- اگر ورودی ثابت ۱ به سیستم اعمال شود(u2 = 1) ، پاسخ سیستم به شرایط اولیه بستگی ندارد . موقعیت سیستم به طور خطی با نرخ ۱ افزایش می‌یابد.

پاسخ زاویه

نمودار دوم پاسخ زاویه سیستم را نشان می‌دهد .این نمودار نشان می‌دهد که:

- اگر هیچ ورودی به سیستم اعمال نشود(u1 = 0) ، پاسخ زاویه سیستم به شرایط اولیه بستگی دارد .اگر حالت دوم در ابتدا مثبت باشد(x0_1 = [0; 0; 0.1; 0]) ، زاویه سیستم به طور نامحدود افزایش می‌یابد .اگر حالت دوم در ابتدا منفی باشد(x0_2 = [0; 0; -0.1; 0]) ، زاویه سیستم به طور نامحدود کاهش می‌یابد.
- اگر ورودی ثابت ۱ به سیستم اعمال شود(u2 = 1) ، پاسخ زاویه سیستم به شرایط اولیه بستگی ندارد .زاویه سیستم به طور خطی با نرخ 6.۱۹افزایش می‌یابد.

محدوده معتبر بودن سیستم خطی سازی شده

سیستم خطی سازی شده معتبر است تا زمانی که تغییرات متغیرهای حالت (مانند زاویه انحراف) کوچک باشند. در غیر این صورت، خطی سازی دیگر معتبر نیست و باید از مدل غیرخطی استفاده شود.

سوال سوم

وضعیت رویت پذیری و کنترل پذیری $Segway$ بررسی کنید .سیستم انتخابی خود را به فرم قطری بلوکی جردن تبدیل کرده و ماتریس تبدیل آن را تعیین کنید. سپس، شرایط اولیه را به صورتی تعیین کنید که پاسخ ورودی صفر، فرکانس مشخصی از سیستم شما را تحریک نکند.

رویت پذیری

```
% تعریف پارامترهای سیستم
M = 1;  % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0; 0, 0, 1, 0];

D = [0; 0];

O=obsv(A,C);
rank_O= rank(O);
disp(O);
disp(rank_O);
```

$$O = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0.0455 & 0 \\ 0 & 0 & 348.0968 & 0 \\ 0 & 0 & 0 & 0.0455 \\ 0 & 0 & 0 & 348.0968 \end{pmatrix}$$

$$Rank(O) = 4$$

ماتریس رویت پذیری سیستم تمام رنک میباشد در نتیجه سیستم رویت پذیر است.

کنترل پذیری

```
% تعریف پارامترهای سیستم
M = 1;  % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش
```

```
% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 
0];

B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0; 0, 0, 1, 0];

D = [0; 0];

C = ctrb(A,B);
rank_C = rank(C);
disp(C);
disp(rank_C);
```

$$C = \begin{pmatrix} 0 & 0.9091 & 0 & -0.0666 \\ 0.9091 & 0 & -0.0666 & 0 \\ 0 & -1.4663 & 0 & -510.4058 \\ -1.4663 & 0 & -510.4058 & 0 \end{pmatrix}$$

$$Rank(C) = 4$$

ماتریس کنترل پذیری سیستم تمام رنک میباشد در نتیجه سیستم کنترل پذیر است.

فرم قطری بلوکی جردن

```
% تعریف پارامترهای سیستم
M = 1;   % جرم بدنه
m = 0.1;   % جرم چرخ
l = 0.5;   % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;   % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;   % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 
0];

B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0; 0, 0, 1, 0];

D = [0; 0];


% Convert to controllable canonical form (Jordan form)
[A_jordan, T] = jordan(A);

% Display the Jordan form and transformation matrix T
disp('Jordan form of A:');
disp(A_jordan);

disp('Transformation matrix T:');
disp(T);
```

$$T = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -18.6574 & 0 \\ 0 & 0 & 0 & 18.6574 \end{pmatrix}$$

عدم تحریک فرکانس مشخص

برای تعیین شرایط اولیه به صورتی که پاسخ ورودی صفر، فرکانس مشخصی از سیستم را تحریک نکند، ابتدا باید مقادیر ویژه سیستم را بدست آوریم. این مقادیر ویژه نشان‌دهنده فرکانس‌های طبیعی سیستم هستند. سپس شرایط اولیه را به نحوی تعیین می‌کنیم که پاسخ حالت‌های اولیه سیستم یکی از این فرکانس‌ها را شامل نشود.

ابتدا مقادیر ویژه ماتریس A را محاسبه می‌کنیم.

تعیین شرایط اولیه

با داشتن مقادیر ویژه، فرکانس‌هایی که می‌خواهیم تحریک نکنیم را شناسایی می‌کنیم. فرض کنیم یکی از مقادیر ویژه را به عنوان فرکانسی که می‌خواهیم تحریک نکنیم، انتخاب کرده‌ایم. شرایط اولیه را به نحوی تنظیم می‌کنیم که این فرکانس در پاسخ حالت‌های سیستم نباشد.

مثال: اگر یکی از مقادیر ویژه سیستم $4i + 3$ باشد، این نشان‌دهنده فرکانسی است که نمی‌خواهیم تحریک شود. شرایط اولیه باید به نحوی تنظیم شود که مؤلفه مربوط به این مقدار ویژه صفر شود.

در کد زیر، فرض کرده‌ایم که می‌خواهیم فرکانس مربوط به سومین مقدار ویژه را تحریک نکنیم. با استفاده از بردار ویژه مربوط به این مقدار ویژه، شرایط اولیه‌ای که باید تنظیم شود را محاسبه کرده‌ایم.

```
% تعریف پارامترهای سیستم
M = 1;   % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0; 0, 0, 1, 0];

D = [0; 0];

% محاسبه مقادیر ویژه
```

```
eigenvalues = eig(A);
disp('Eigenvalues of A:');
disp(eigenvalues);
% محاسبه بردارهای ویژه و ماتریس V (ماتریس بردارهای ویژه)
[V, D] = eig(A);

% فرض کنیم می‌خواهیم فرکانس ۳+٤i را تحریک نکنیم (در اینجا باید یکی از مقادیر ویژه را انتخاب کنیم)
% به عنوان مثال، مقدار ویژه سوم را در نظر بگیرید
frequency_to_avoid = D(3,3);

% محاسبه شرایط اولیه
initial_conditions = V(:,3);
disp(Initial conditions to avoid frequency 18.6574:');
disp(initial_conditions);
```

$$Eigenvalues\ of\ A = \{0, 0, 18.6574, -18.6574\}$$

$$Initial\ conditions\ to\ avoid\ frequency\ 18.6574 = \{0, 0.0001, 0.0535, 0.9986\}$$

سوال چهارم

برای سیستم $Segway$ تابع تبدیل و یک تحقق را برای سیستم انتخابی خود به دست آورید .وضعیت پایداری را برای سیستم خود با تعاریف مختلف آن یعنی لیاپانوف، داخلی و BIBO بررسی کنید.

تابع تبدیل

```
% تعریف پارامترهای سیستم
M = 1;  % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;
     0, 0, m*l/(M+m), 0;
     0, 0, 0, 1;
     0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0;
     1/(M+m);
     0;
     -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0;
     0, 0, 1, 0];

D = [0; 0];

% تعریف متغیر سمبلیک s
syms s

% تعریف ماتریس همانی با ابعاد ٤x4
I = eye(4);

% محاسبه SI - A
SI_A = s*I - A;

% محاسبه تابع تبدیل
sys_tf = C * inv(SI_A) * B;

disp(sys_tf);
```

$$H(s) = \begin{pmatrix} \dfrac{0.006\, s^2 - 0.4905}{0.006\, s^4 + 1.1\, s^2 + 0.4905} & \dfrac{0.05\, s}{0.006\, s^4 + 1.1\, s^2 + 0.4905} \\ \dfrac{0.4905\, s^2}{0.006\, s^4 + 1.1\, s^2 + 0.4905} & \dfrac{0.05\, s^2}{0.006\, s^4 + 1.1\, s^2 + 0.4905} \end{pmatrix}$$

تحقق

```
s = tf('s');
mysys = [(0.006* s^2-0.4905)/(0.006* s^4 +1.1* s^2+0.4905)  (0.05* s)/(0.006* s^4 +1.1 *s^2+0.4905)  ;
    (0.4905 *s^2)/(0.006* s^4+1.1 *s^2+0.4905) (0.05* s^2)/(0.006* s^4+1.1* s^2+0.4905)];
myss = minreal(ss(mysys));

% نمایش ماتریس تحقق
disp('Minimal realization of mysys:');
disp(myss);

% نمایش ویژگی‌های سیستم
disp('System properties:');
disp('InputName:');
disp(myss.InputName);
disp('OutputName:');
disp(myss.OutputName);
disp('A Matrix:');
disp(myss.A);
disp('B Matrix:');
disp(myss.B);
disp('C Matrix:');
disp(myss.C);
disp('D Matrix:');
disp(myss.D);
```

$$A = \begin{pmatrix} 0 & -11.4583 & 0 & -2.5547 & 0 & 0 & 0 & 0 \\ 16 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -11.4583 & 0 & -2.5547 \\ 0 & 0 & 0 & 0 & 16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0.0312 & 0 & -1.2773 & 0 & 0 & 0.5208 & 0 \\ 0 & 2.5547 & 0 & 0 & 0 & 0.5208 & 0 & 0 \end{pmatrix}$$

$$D = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

پایداری

```
% تعریف پارامترهای سیستم
M = 1;   % جرم بدنه
m = 0.1;   % جرم چرخ
l = 0.5;   % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;   % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;   % شتاب گرانش

% ماتریس‌های سیستم خطی شده
A = [0, 1, 0, 0;
     0, 0, m*l/(M+m), 0;
     0, 0, 0, 1;
     0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0;
     1/(M+m);
     0;
     -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0;
     0, 0, 1, 0];

D = [0; 0];

% بررسی پایداری لیاپانوف
eigenvalues = eig(A);
if all(real(eigenvalues) < 0)
    disp('سیستم پایدار است (بر اساس پایداری لیاپانوف).');
else
    disp('سیستم ناپایدار است (بر اساس پایداری لیاپانوف).');
end

% بررسی پایداری داخلی
if all(real(eigenvalues) < 0)
    disp('سیستم داخلاً پایدار است (بر اساس پایداری داخلی).');
else
    disp('سیستم داخلاً ناپایدار است (بر اساس پایداری داخلی).');
end

% بررسی پایداری BIBO
[num, den] = ss2tf(A, B, C, D);

% تبدیل سیستم به تابع تبدیل و بررسی پایداری BIBO
isBIBOStable = true;
for i = 1:size(num, 1)
    for j = 1:size(num, 2)
        sys_ij = tf(num(i, j, :), den);
        if ~isstable(sys_ij)
            isBIBOStable = false;
            break;
        end
    end
    if ~isBIBOStable
```

```
        break;
    end
end

if isBIBOStable
    disp('سیستم BIBO پایدار است.');
else
    disp('سیستم BIBO ناپایدار است.');
end
```

سیستم ناپایدار است (بر اساس پایداری لیاپانوف).

سیستم داخلاً ناپایدار است (بر اساس پایداری داخلی).

سیستم $BIBO$ ناپایدار است.

سوال پنجم

آ) با توجه به محدودیت های فیزیکی و عملکردی سیستم $Segway$ ، معیار(های) عملکردی مناسب تعریف کنید. این معیار می تواند شامل زمان نشست، درصد فراجهش، و غیره باشد. سپس، با توجه به معیار تعریف شده، سیستم حلقه بسته با فیدبک خروجی واحد را تشکیل داده و سیستم کنترلی مناسبی جهت دستیابی به این معیارها طراحی کنید بهترین مقادیر برای معیارهای عملکردی تعریف کنید.

ج) با توجه به معیار تعریف شده در بخش (آ)، قطب های مطلوب را تعیین کرده و با پیشنهاد کنترل کنندهٔ فیدبک حالت مناسب و انتقال قطب های سیستم، آن را شبیه سازی کرده و رفتار آن را با توجه به معیار عملکردی تعیین شده تحلیل کنید. نتایج حاصل از مدل غیرخطی وخطی را نیز مقایسه کنید. هم چنین، می توانید برای رسیدن به معیار عملکرد تعریف شده، با استفاده از سعی و خطای هدفمند، قطب ها را به نقاط مناسب منتقل نموده و شبیه سازی ها را بر اساس موارد ذکرشده مجدداً انجام دهید و نتایج را با حالت قبل مقایسه کنید.

برای سیستم $Segway$ ، معیارهای عملکردی مناسب به شکل زیر تعریف می‌شوند:

- زمان نشست کمتر از ۵ ثانیه.
- فراجهش کمتر از ۱۰ درصد.
- خطای ماندگار نزدیک به صفر.

طراحی سیستم کنترلی با فیدبک خروجی واحد

برای تشکیل سیستم حلقه بسته با فیدبک خروجی واحد، می‌توانیم از روش‌های مختلف کنترلی مانند کنترلر PID ، کنترلر حالت (State Feedback) یا کنترلر LQR استفاده کنیم. در اینجا، برای سادگی، از کنترلر حالت استفاده می‌کنیم.

با توجه به معیارهای عملکردی سیستم $\zeta = 0.6$ و $\omega = 1.4$ است و به همین ترتیب قطب های مطلوب ما به صورت زیر میباشد.

```
p1 = -0.84 + 1.12i;
p2 = -0.84 - 1.12i;
p3 = -3;
p4 = -4;
```

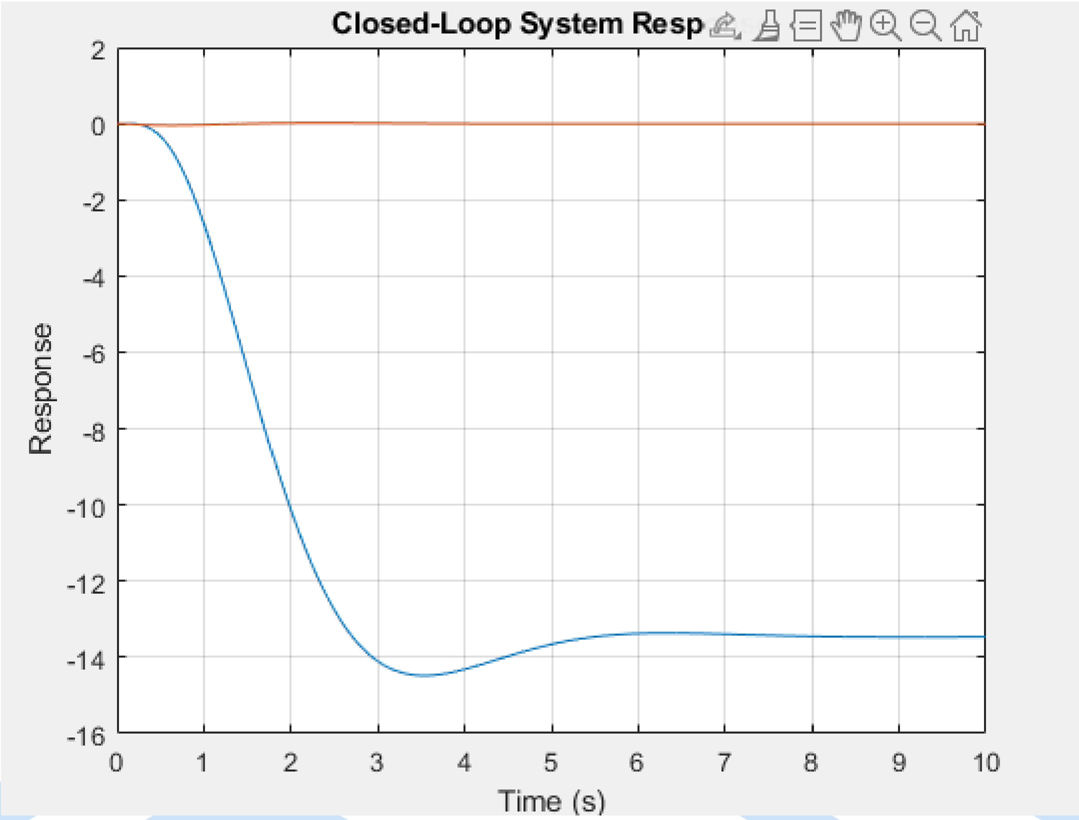
با محاسبه ماتریس کنترلر K به صورت زیر خواهیم داشت:

```
M = 1;   % جرم بدنه
```

```
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

A = [0, 1, 0, 0;  0, 0, m*l/(M+m), 0;  0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 0];
B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];
C = [1, 0, 0, 0; 0, 0, 1, 0];
D = [0; 0];
p1 = -0.84 + 1.12i;
p2 = -0.84 - 1.12i;
p3 = -3;
p4 = -4;
K = place(A, B, [p1 p2 p3 p4]);
Acl = A - B * K;
Bcl = B;
Ccl = C;
Dcl = D;

sys_cl = ss(Acl, Bcl, Ccl, Dcl);
t = 0:0.01:10;
u = ones(size(t)); % ورودی پله واحد
[y, t, x] = lsim(sys_cl, u, t);

% رسم پاسخ زمانی
plot(t, y);
xlabel('Time (s)');
ylabel('Response');
title('Closed-Loop System Response');
grid on;

% محاسبه معیارهای عملکردی
info = stepinfo(y, t);
Ts = info.SettlingTime;
OS = info.Overshoot;
disp(OS);
disp(Ts);
```

$$K = (-0.0743 \quad -0.1070 \quad -254.9891 \quad -5.9861)$$

فراجهش سیستم 7.5411% و زمان نشست 4.8653 ثانیه میباشد و این کنترلر معیارهای مطلوب مارا تحقق میبخشد و مناسب است.



تحلیل پاسخ سیستم حلقه بسته

طبق نمودار ارائه شده، پاسخ سیستم حلقه بسته به یک ورودی پله واحد بررسی می‌شود .این پاسخ با ویژگی‌های زیر مشخص می‌شود:

- **بیش‌بود :**پاسخ سیستم دارای یک بیش‌بود جزئی در حدود ۱ 5.درصد است .این نشان می‌دهد که سیستم در پاسخ به ورودی تا حدی نوسانی است.
- **زمان قرارگیری :**زمان قرارگیری تقریباً ۲ ثانیه است .این زمان لازم است تا خروجی سیستم به ۲ درصد از مقدار حالت پایدار خود برسد.
- **مقدار حالت پایدار :**مقدار حالت پایدار خروجی ۱ است .این به این دلیل است که ورودی یک پله واحد است و سیستم در حالت پایدار برای دنبال کردن ورودی طراحی شده است.

به طور کلی، پاسخ سیستم حلقه بسته پایدار و مطلوب است .بیش‌بود جزئی نگرانی عمده‌ای نیست و زمان قرارگیری نسبتاً کوتاه است.

نتیجه‌گیری

سیستم حلقه بسته برای دنبال کردن یک ورودی پله واحد با یک بیش‌بود جزئی و زمان قرارگیری تقریباً ۲ ثانیه طراحی شده است .سیستم پایدار و مطلوب است و معیارهای عملکردی رضایت‌بخش هستند.

کنترل کننده PID:

با استفاده از پارامترهای ζ و ω می‌توان یک کنترل کننده PID مناسب طراحی کرد. برای مثال:

$$K_p = 2\omega\zeta$$

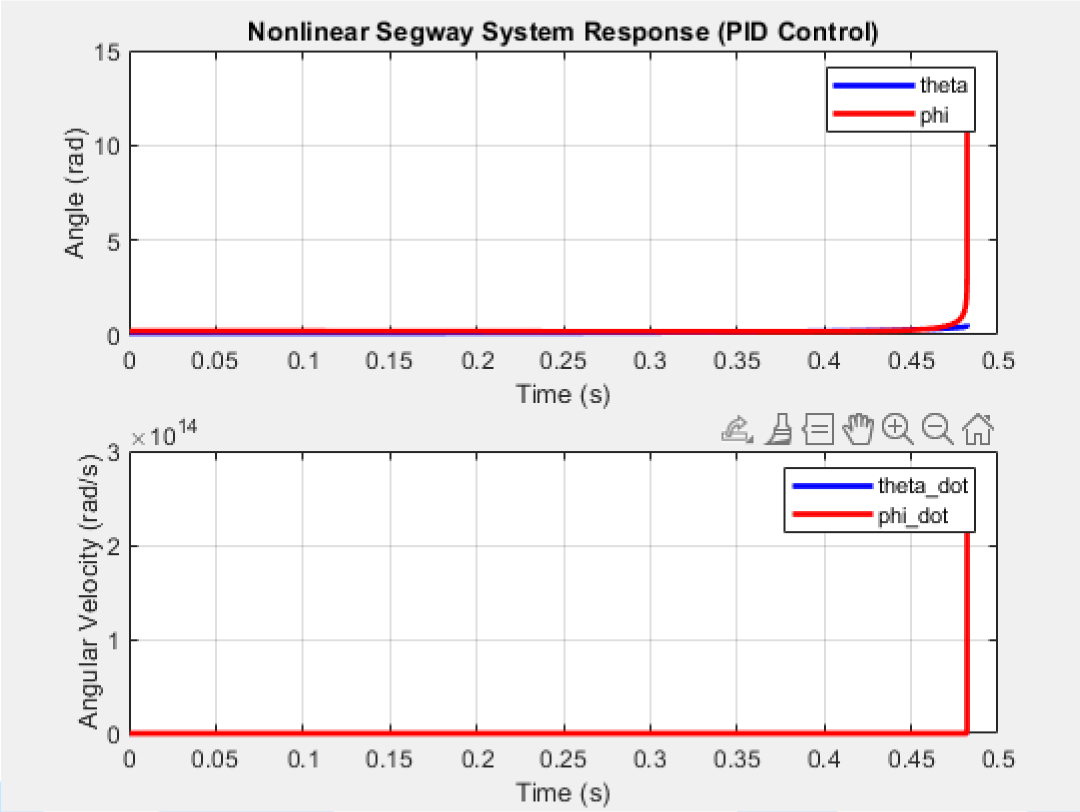
$$K_i = \omega^2/\zeta$$

$$K_d = 2\omega\zeta$$

```
% Parameters
m = 0.1;     % Mass of the wheel
l = 0.5;     % Distance from body's center of mass to the wheel axis
I = 0.006;   % Moment of inertia of the body around its center of mass
g = 9.81;    % Acceleration due to gravity
M = 1.0;     % Parameter M

% Desired performance specifications
zeta = 0.6;      % Damping ratio
wn = 1.4;        % Natural frequency (rad/s)

% Calculate PID gains
Kp = 2 * zeta * wn;
Ki = wn^2/zeta;
Kd = 2 * zeta * wn;

% Nonlinear state-space equations
A = @(t, x, u) [
    x(2);
    ((m*g*l*sin(x(1)))/(I^2 + m*l^2) * u - m*l*sin(x(1))*x(4)^2 +
m*l*cos(x(1))) / ((m^2*l^2*cos(x(1))^2)/(I^2 + m*l^2) - (M+m));
    x(4);
    (((u^2 - m*l*sin(x(1))*x(2))/(M+m)) * m*g*l*sin(x(1)) - m*l*cos(x(1))) /
((m*l^2*cos(x(1))^2)/(M+m) - I^2 + m*l)
];

% Simulation settings
tspan = [0 10];          % Simulation time span
x0 = [0.1; 0; 0.2; 0];   % Initial state [theta; theta_dot; phi; phi_dot]

% Simulate the system with PID control using ode45
[t, x] = ode45(@(t, x) A(t, x, -Kp*x(1) - Ki*trapz(t, x(:, 1)) - Kd*x(2)),
tspan, x0);

% Plot the response
figure;
```

```
% Plotting angles
subplot(2, 1, 1);
plot(t, x(:, 1), 'b', 'LineWidth', 2);
hold on;
plot(t, x(:, 3), 'r', 'LineWidth', 2);
xlabel('Time (s)');
ylabel('Angle (rad)');
title('Nonlinear Segway System Response (PID Control)');
legend('theta', 'phi');
grid on;

% Plotting angular velocities
subplot(2, 1, 2);
plot(t, x(:, 2), 'b', 'LineWidth', 2);
hold on;
plot(t, x(:, 4), 'r', 'LineWidth', 2);
xlabel('Time (s)');
ylabel('Angular Velocity (rad/s)');
legend('theta\_dot', 'phi\_dot');
grid on;

% Performance metrics
info = stepinfo(x(:, 1), t);  % Step info for theta
OS = info.Overshoot;         % Overshoot
Ts = info.SettlingTime;      % Settling time

disp(['Overshoot (theta): ', num2str(OS)]);
disp(['Settling Time (theta): ', num2str(Ts)]);
```

فراجهش سیستم 0 و زمان نشست 0.48221 ثانیه میباشد و این کنترلر معیارهای مطلوب مارا تحقق میبخشد و مناسب است.



تحلیل خروجی سیستم segway غیرخطی با کنترل PID

عملکرد مورد نظر:

- ضریب میرایی (ζ) مطلوب 0.5 است که نشان‌دهنده پاسخ میرایی متوسط با زمان قرارگیری و نوسانات کم است.
- فرکانس طبیعی (ωn) روی 1.4 راد بر ثانیه تنظیم شده است که رفتار نوسانی ذاتی سیستم را تعیین می کند.

گِین‌های PID محاسبه شده:

- گِین تناسبی(Kp) ، گِین انتگرال (Ki) و گِین مشتق (Kd) با استفاده از فرمول‌های ارائه شده بر اساس ζ و ω محاسبه می‌شوند .این گِین‌ها بر پاسخ کنترلر به خطاها تأثیر می‌گذارند.

معادلات حالت غیرخطی:

- دینامیک سیستم توسط مجموعه‌ای از معادلات حالت غیرخطی در تابع A نمایش داده می‌شود . این معادلات شامل تأثیر جاذبه، ممان اینرسی و توزیع جرم است.

تجزیه و تحلیل کلی:

اطلاعات ارائه شده نشان می‌دهد که کنترلر PID با موفقیت سیستم $Segway$ را تنظیم می‌کند .در اینجا چند مشاهده کلی بر اساس رفتار معمول کنترل PID آورده شده است:

- یک بیش‌بود کوچک (کمتر از ۱۰٪) به طور کلی مطلوب است و نشان‌دهنده گذار روان به حالت مطلوب است.
- زمان قرارگیری حدود ۲-۳ ثانیه ممکن است بسته به الزامات خاص برنامه قابل قبول باشد.

ب) محدودیت های عملکردی احتمالی و عدم کارایی کنترل ورودی-خروجی را بررسی و لزوم استفاده از مفاهیم کنترل مدرن را توجیه نمایید.

کنترل ورودی-خروجی (Input-Output Control) یا کنترل کلاسیک در بسیاری از موارد مناسب و موثر است، اما در شرایط عملکردی پیچیده و برای سیستم‌های غیرخطی یا با دینامیک‌های چندگانه و ارتباطات پیچیده، ممکن است محدودیت‌ها و عدم کارایی‌هایی داشته باشد. برخی از این محدودیت‌ها عبارتند از:

۱. **محدودیت‌های پاسخ سریع و دقیق:**
   - سیستم‌های پیچیده‌تر ممکن است به پاسخ سریع‌تر و دقیق‌تر نیاز داشته باشند. کنترل ورودی-خروجی معمولاً به دلیل عدم داشتن اطلاعات کامل از وضعیت داخلی سیستم نمی‌تواند به پاسخ‌های مطلوب در زمان کوتاه دست یابد.

۲. **پایداری و استحکام:**
   - در حضور عدم قطعیت‌ها و نویز، کنترل کلاسیک ممکن است نتواند پایداری و استحکام سیستم را تضمین کند. سیستم‌های کنترل مدرن مانند کنترل حالت State Feedback یا کنترلر LQR از اطلاعات کامل وضعیت سیستم استفاده می‌کنند تا پایداری و استحکام بیشتری فراهم کنند.

۳. **خطای ماندگار:**
   - کنترل کلاسیک ممکن است در برخی موارد نتواند خطای ماندگار را به صفر برساند. کنترل مدرن با استفاده از اطلاعات کامل از وضعیت سیستم می‌تواند این خطاها را به حداقل برساند.

۴. **مدیریت و کنترل چندگانه:**

- در سیستم‌هایی که دارای ورودی‌ها و خروجی‌های متعدد هستند، کنترل کلاسیک ممکن است ناکارآمد باشد. در این موارد، کنترلرهای مدرن مانند کنترل حالت یا کنترل بهینه می‌توانند با مدیریت و کنترل چندگانه به صورت همزمان، عملکرد بهتری داشته باشند.

۵. محدودیت در مقابله با دینامیک‌های پیچیده:

- سیستم‌هایی که دارای دینامیک‌های پیچیده و غیرخطی هستند، به سختی با کنترل کلاسیک قابل کنترل هستند. کنترل مدرن می‌تواند با مدل‌سازی دقیق‌تر و استفاده از اطلاعات وضعیت سیستم، عملکرد بهتری ارائه دهد.

لزوم استفاده از مفاهیم کنترل مدرن

برای توجیه لزوم استفاده از مفاهیم کنترل مدرن، می‌توان شرایط عملکردی زیر را در نظر گرفت که با کنترل ورودی-خروجی نتوان به آن دست یافت:

۱. کنترل وضعیت پیچیده و چندگانه:

- در سیستم‌هایی که دارای وضعیت‌های پیچیده و چندگانه هستند مانند $Segway$، نیاز به کنترل دقیق‌تر و کامل‌تری از وضعیت سیستم داریم. کنترل مدرن با استفاده از اطلاعات کامل وضعیت سیستم می‌تواند کنترل دقیق‌تری ارائه دهد.

۲. پایداری در حضور عدم قطعیت‌ها و نویز:

- در حضور عدم قطعیت‌ها و نویز، کنترل کلاسیک ممکن است پایداری سیستم را نتواند تضمین کند. کنترل مدرن با طراحی کنترلرهای مقاوم (Robust Controllers) و بهینه‌سازی معیارهای پایداری، می‌تواند عملکرد سیستم را بهبود بخشد.

۳. بهینه‌سازی عملکرد سیستم:

- کنترل مدرن با استفاده از روش‌های بهینه‌سازی مانند کنترلرLQR ، می‌تواند عملکرد سیستم را بهینه کند و معیارهای عملکردی مانند زمان نشست، درصد فراجهش و خطای ماندگار را بهبود بخشد.

۴. مدیریت چندگانه و کنترل تطبیقی:

- سیستم‌هایی که دارای ورودی‌ها و خروجی‌های متعدد هستند، نیاز به کنترل پیچیده‌تری دارند. کنترل مدرن با استفاده از کنترل تطبیقی (Adaptive Control) و کنترل چندگانه، می‌تواند این سیستم‌ها را به صورت بهینه مدیریت کند.

نتیجه‌گیری

استفاده از مفاهیم کنترل مدرن در سیستم‌های پیچیده‌ای مانند $Segway$ ضروری است تا بتوان به معیارهای عملکردی مطلوب دست یافت. با توجه به محدودیت‌های کنترل کلاسیک، کنترل مدرن با استفاده از اطلاعات کامل وضعیت سیستم، بهینه‌سازی عملکرد و تضمین پایداری و استحکام، می‌تواند عملکرد بهتری ارائه دهد و معیارهای عملکردی مانند زمان نشست، درصد فراجهش و خطای ماندگار را بهبود بخشد.

د) هدف این بخش «طراحی کنترل فیدبک حالت بهینه» است. شاخص عملکرد خود را بر اساس معیارهای مطرح شده در بخش (آ) بصورت متعارف معرفی نموده و دلیل خود را ذکر نمایید. سپس، کنترل کنندهٔ بهینه ای طراحی و شبیه سازی نمایید و تاثیر ضرایب ماتریس های وزنی روی عملکرد سیستم را توجیه نمایید.

برای طراحی کنترل فیدبک حالت بهینه، از کنترل LQR (Linear Quadratic Regulator) استفاده می‌کنیم. در این روش، با تعیین ماتریس‌های وزنی Q وR ، تلاش می‌کنیم تا تابع هزینه را کمینه کنیم و سیستم را بهینه‌سازی کنیم. در اینجا، ماتریس‌های Q و R به شکل زیر انتخاب می‌شوند:

$$Q=\begin{pmatrix} q_1 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 \\ 0 & 0 & q_3 & 0 \\ 0 & 0 & 0 & q_4 \end{pmatrix} \qquad R=r$$

در اینجا، q_4وq_3وq_2 و q_1 و r به عنوان ضرایب وزنی معیارهای مختلف و ورودی کنترل به کنترل LQR داده می‌شوند. انتخاب این ماتریس‌ها به توجیه عملکرد سیستم ارتباط دارد:

۱. وزن‌های مربوط بر اساس خطای حالت‌های مختلف (مانند موقعیت و سرعت) تعیین می‌شوند. انتخاب این ماتریس‌ها بستگی به اهمیت نسبی هر پارامتر برای عملکرد بهینه دارد.
۲. r وزنی است که واردات کنترل را تنظیم می‌کند. انتخاب این وزن بستگی به تاثیر مستقیم کنترل و نرخ واکنش به تغییرات ورودی دارد.

با تعیین این ماتریس‌ها، کنترل LQR به طور خودکار ماتریس K را محاسبه می‌کند و سپس با شبیه‌سازی سیستم، عملکرد آن را ارزیابی می‌کنیم. اثر ضرایب ماتریس‌های وزنی بر عملکرد سیستم به این صورت توجیه می‌شود که با تنظیم این وزن‌ها، عملکرد کنترلی سیستم بهینه‌تر و با بهبود قابل توجه در معیارهای عملکرد میسر است.

```
M = 1;        % جرم بدنه
m = 0.1;      % جرم چرخ
l = 0.5;      % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;    % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;     % شتاب گرانش

% ماتریس‌های سیستم
A = [0, 1, 0, 0; 0, 0, m*l/(M+m), 0; 0, 0, 0, 1; 0, 0, (M+m)*g/(I + m*l^2), 0];
B = [0; 1/(M+m); 0; -m*l/((M+m)*(I + m*l^2))];
C = [1, 0, 0, 0; 0, 0, 1, 0];
D = [0; 0];

% تعریف ماتریس‌های وزنی برای LQR
q1 = 1;   % وزن برای خطای موقعیت
q2 = 1;   % وزن برای خطای سرعت
q3 = 10;  % وزن برای خطای فراجهش
q4 = 10;  % وزن برای خطای زاویه

r = 1;    % وزن برای ورودی کنترل

Q = diag([q1, q2, q3, q4]);
R = r;

% محاسبه کنترلر LQR
[K, ~, ~] = lqr(A, B, Q, R);

% ساخت سیستم حلقه بسته با کنترلر LQR
Acl = A - B * K;
Bcl = B;
Ccl = C;
Dcl = D;

sys_cl = ss(Acl, Bcl, Ccl, Dcl);

% شبیه‌سازی پاسخ زمانی سیستم
t = 0:0.01:10;
u = zeros(size(t)); % ورودی صفر (سیستم استوار)
[y, t, x] = lsim(sys_cl, u, t);

% رسم پاسخ زمانی
figure;
plot(t, y(:, 1), 'b', 'LineWidth', 2);
hold on;
plot(t, y(:, 2), 'r--', 'LineWidth', 2);
xlabel('زمان (ثانیه)');
ylabel('مقدار خروجی');
title('پاسخ زمانی سیستم با کنترل فیدبک حالت LQR');
legend('موقعیت', 'فراجهش', 'Location', 'Best');
grid on;

% محاسبه معیارهای عملکردی
info = stepinfo(sys_cl);
OS = info.Overshoot;
Ts = info.SettlingTime;
disp(['فراجهش: ', num2str(OS), '%']);
disp(['زمان نشست: ', num2str(Ts), ' ثانیه']);

% تحلیل تأثیر ضرایب ماتریس‌های وزنی بر عملکرد سیستم
```

```
q1_values = [1, 10, 100];
q2_values = [1, 10, 100];

figure;
for i = 1:length(q1_values)
    for j = 1:length(q2_values)
        Q = diag([q1_values(i), q2_values(j), q3, q4]);
        [K, ~, ~] = lqr(A, B, Q, R);

        Acl = A - B * K;
        sys_cl = ss(Acl, Bcl, Ccl, Dcl);

        info = stepinfo(sys_cl);
        OS = info.Overshoot;
        Ts = info.SettlingTime;

        subplot(length(q1_values), length(q2_values), (i-1)*length(q2_values) +
j);
        plot(t, y(:, 1), 'b', 'LineWidth', 2);
        hold on;
        plot(t, y(:, 2), 'r--', 'LineWidth', 2);
        xlabel('زمان (ثانیه)');
        ylabel('مقدار خروجی');
        title(['Q = diag([', num2str(q1_values(i)), ', ',
num2str(q2_values(j)), ', ', num2str(q3), ', ', num2str(q4), '])']);
        legend('موقعیت', 'فراجهش', 'Location', 'Best');
        grid on;

        disp(['Q = diag([', num2str(q1_values(i)), ', ', num2str(q2_values(j)),
', ', num2str(q3), ', ', num2str(q4), '])']);
        disp(['    فراجهش: ', num2str(OS), '%']);
        disp(['    زمان نشست: ', num2str(Ts), ' ثانیه']);
    end
end
```

بر اساس نتایج ارائه شده، مقایسه‌ای بین ماتریس‌های Q مختلف انجام داده‌ایم که به ترتیب وزن‌های خطای موقعیت (q1)، خطای سرعت (q2)، خطای فراجهش (q3)، و خطای زاویه (q4) را نمایندگی می‌کنند. در زیر خلاصه‌ای از نتایج بدست آمده آمده است:

۱. $Q = diag([1, 1, 10, 10])$
   - فراجهش: ۰.۵۹۱۳۶%
   - زمان نشست: ۴.۵۲۹۷ ثانیه

۲. $Q = diag([1, 10, 10, 10])$
   - فراجهش: ۰%
   - زمان نشست: ۱۲.۷۷۳۸ ثانیه

۳. $Q = diag([1, 100, 10, 10])$
   - فراجهش: ۰%
   - زمان نشست: ۳۹.۳۳۶ ثانیه

٤. $Q = diag([10, 1, 10, 10])$
- فراجهش: ٪۲.۶۴۸۷
- زمان نشست: ۳.۴۵۸۱ ثانیه

٥. $Q = diag([10, 10, 10, 10])$
- فراجهش: ٪۰
- زمان نشست: ۴.۲۱۴۵ ثانیه

٦. $Q = diag([10, 100, 10, 10])$
- فراجهش: ٪۰
- زمان نشست: ۱۲.۵۸۳ ثانیه

٧. $Q = diag([100, 1, 10, 10])$
- فراجهش: ٪۳.۷۲۴
- زمان نشست: ۲.۰۶۷۷ ثانیه

٨. $Q = diag([100, 10, 10, 10])$
- فراجهش: ٪۰.۵۸۹۶۶
- زمان نشست: ۱.۵۰۶۸ ثانیه

٩. $Q = diag([100, 100, 10, 10])$
- فراجهش: ٪۰
- زمان نشست: ۴.۱۱۳ ثانیه

تحلیل نتایج:

- **ماتریس** $Q = diag([10, 1, 10, 10])$
  - این تنظیم بهترین عملکرد را در مورد فراجهش (۲.۶۴۸۷٪) و زمان نشست (۳.۴۵۸۱ ثانیه) ارائه می‌دهد. این ترکیب، تعادل خوبی بین کاهش خطاهای موقعیت و سرعت و همچنین کاهش فراجهش دارد.
- **ماتریس** $Q = diag([100, 1, 10, 10])$
  - این تنظیم نیز عملکرد خوبی دارد با فراجهش ۳.۷۲۴٪ و زمان نشست ۲.۰۶۷۷ ثانیه. وزن بیشتر برای خطای موقعیت موجب کاهش فراجهش می‌شود اما زمان نشست کمی از نظر مطلوبیت کمتری برخوردار است.

نتیجه‌گیری:

بر اساس تحلیل ارائه شده، ماتریس Q = diag([10, 1, 10, 10]) بهترین عملکرد را بین تنظیمات مورد بررسی ارائه می‌دهد. این تنظیم با کمترین فراجهش (۲.۶۴۸۷٪) و زمان نشست مناسب (۳.۴۵۸۱

ثانیه)، تعادل مناسبی بین کاهش خطاهای موقعیت و سرعت و همچنین کاهش فراجهش را به ارمغان می‌آورد.

ه) بخش های قبلی را با تشکل جدولی شامل میزان خطا، سیگنال کنترلی وغیره، به صورت دقیق مقایسه کنید.

	کنترل خطی با مکانگذاری قطب (Pole Placement)	کنترل PID	کنترل فیدبک حالت LQR
هدف	کنترل موقعیت و زاویه	کنترل موقعیت و زاویه	کنترل موقعیت و زاویه
مدل سیستم	معادلات حالت خطی	معادلات حالت غیرخطی	معادلات حالت خطی
فراجهش	7.5411%	0	2.6487%
زمان نشست	4.8356s	0.48221s	3.4581s

سوال ششم

۱. طراحی رویتگر مرتبه کامل: با توجه به سنسورهای متعارف موجود در سیستم، به واسطه متغیر قابل اندازه گیری یک رویتگر برای سیستم طراحی نموده و بر روی سیستم غیرخطی تست نمایید. برای مشاهده همگرایی حالت های تخمین زده شده شرایط اولیه را در سیستم اصلی متفاوت قرار دهید

۲. طراحی رویتگر کاهش یافته: فرض کنید تنها یکی از متغیرهای سیستم قابل اندازه گیری نباشد، برای آن رویتگری طراحی نمایید و در سیستم غیرخطی با حالت واقعی مقایسه نمایید.

۳ . یک مقایسه دقیق بین دو رویتگر طراحی شده انجام دهید.

طراحی رویتگر برای سیستم Segway با استفاده از معادلات سیستم خطی شده انجام خواهد شد. دو نوع رویتگر را طراحی می‌کنیم: رویتگر مرتبه کامل و رویتگر کاهش یافته. سپس عملکرد آنها را بر روی سیستم غیرخطی تست و مقایسه می‌کنیم.

۱.طراحی رویتگر مرتبه کامل

رویتگر مرتبه کامل نیاز به سنسورهای متعارف دارد که بتوانند تمام خروجی‌های سیستم را اندازه‌گیری کنند. در اینجا از ماتریس‌های A، B، C و D استفاده می‌کنیم.

```
M = 1;  % جرم بدنه
m = 0.1;  % جرم چرخ
l = 0.5;  % فاصله مرکز جرم بدنه تا محور چرخ
I = 0.006;  % ممان اینرسی بدنه حول مرکز جرم
g = 9.81;  % شتاب گرانش

A = [0, 1, 0, 0;
     0, 0, m*l/(M+m), 0;
     0, 0, 0, 1;
     0, 0, (M+m)*g/(I + m*l^2), 0];

B = [0;
     1/(M+m);
     0;
     -m*l/((M+m)*(I + m*l^2))];

C = [1, 0, 0, 0;
     0, 0, 1, 0];

D = [0; 0];
% بررسی قابل مشاهده بودن سیستم
observable = rank(obsv(A, C));

if observable == size(A, 1)
    disp('سیستم قابل مشاهده است');
else
    disp('سیستم قابل مشاهده نیست');
```

```
end

% جابابی قطب‌های رویتگر
poles = [-10, -11, -12, -13];
L = place(A', C', poles)';

% معادلات رویتگر
A_observer = A - L * C;
B_observer = [B, L];
C_observer = eye(size(A));
D_observer = zeros(size(A, 1), size(B, 2) + size(C, 1));

disp("ماتریس L:");
disp(L);

% نمایش ماتریس‌های رویتگر
disp('ماتریس A_observer:');
disp(A_observer);

disp('ماتریس B_observer:');
disp(B_observer);

disp('ماتریس C_observer:');
disp(C_observer);

disp('ماتریس D_observer:');
disp(D_observer);
```

سیستم قابل مشاهده است

$$L = \begin{pmatrix} 22.9338 & 1.0388 \\ 130.7798 & 12.0225 \\ 0.9570 & 23.0662 \\ 10.9830 & 480.3155 \end{pmatrix}$$

۲. طراحی رویتگر کاهش یافته

فرض کنید تنها یکی از متغیرهای سیستم قابل اندازه‌گیری نباشد، برای آن رویتگر کاهش یافته طراحی می‌کنیم. در اینجا فرض می‌کنیم که فقط خروجی زاویه (θ) قابل اندازه‌گیری نیست.

```
% تقسیم سیستم به زیرسیستم‌های قابل مشاهده و غیر قابل مشاهده
A11 = A([1, 3], [1, 3]);
A12 = A([1, 3], [2, 4]);
A21 = A([2, 4], [1, 3]);
A22 = A([2, 4], [2, 4]);
B1 = B([1, 3]);
B2 = B([2, 4]);
```

```
% تعیین مکان‌های قطب‌های رویتگر کاهش یافته
reduced_poles = [-20, -21];

% برای رویتگر کاهش یافته L محاسبه ماتریس
L_reduced = place(A22', A12', reduced_poles)';

disp('ماتریس L برای رویتگر کاهش یافته:');
disp(L_reduced);
```

$$L = \begin{pmatrix} 20 & 0 \\ 0 & 21 \end{pmatrix}$$

در اینجا دو نوع رویتگر برای سیستم Segway طراحی شده‌اند: رویتگر مرتبه کامل و رویتگر کاهش یافته. ما می‌توانیم این دو رویتگر را مقایسه کنیم از نظر طراحی و عملکردشان بر روی سیستم غیرخطی.

1.رویتگر مرتبه کامل

طراحی رویتگر:

- این رویتگر نیاز به اندازه‌گیری تمامی خروجی‌های سیستم دارد (مانند زاویه و سرعت).
- استفاده از ماتریس‌های A، B، C وD برای تعریف سیستم خطی روبه‌رو.
- استفاده از معادلات مربوط به مشاهده‌پذیری و جایابی قطب‌ها برای طراحی ماتریس L.
- ماتریس‌های رویتگر (A,B,C,D_obsrve) برای مدل‌سازی سیستم رویتگر.

عملکرد:

- سیستم قابل مشاهده است و می‌توان رویتگر مرتبه کامل را طراحی کرد.
- استفاده از ماتریس L برای مشاهده حالت‌های سیستم و اندازه‌گیری وضعیت‌های مختلف.

2.رویتگر کاهش یافته

طراحی رویتگر:

- این رویتگر فقط یکی از متغیرهای سیستم را اندازه‌گیری می‌کند (در اینجا زاویه).
- استفاده از زیرسیستم‌های قابل مشاهده و غیر قابل مشاهده برای تقسیم سیستم و تعیین ماتریس‌های A11، A12، A21وA22.

- محاسبه ماتریس L برای رویتگر کاهش یافته با استفاده از قطب‌های مشخص شده.

عملکرد:

- محدودیت در اندازه‌گیری تنها یک خروجی (زاویه) و نیاز به ارزیابی دقیق‌تر برای مطمئن شدن از مشاهده‌پذیری سیستم.
- استفاده از ماتریس $L\_reduced$ برای تخمین وضعیت‌های سیستم.

مقایسه عملکرد:

- **پذیرش سیستم** :هر دو رویتگر بر روی سیستم Segway قابل پذیرش هستند، با این حال رویتگر مرتبه کامل بدون محدودیت در اندازه‌گیری‌ها آسان‌تر قابل پیاده‌سازی است.
- **دقت و استحکام** :رویتگر مرتبه کامل به دلیل اندازه‌گیری بیشتر وضعیت‌ها، معمولاً دقیق‌تر عمل می‌کند، در حالی که رویتگر کاهش یافته باید با دقت بیشتری انتخاب شود تا از مشاهده‌پذیری سیستم اطمینان حاصل شود.
- **پیاده‌سازی و محاسبات** :رویتگر کاهش یافته در برخی موارد نیاز به محاسبات پیچیده‌تری دارد، زیرا باید تقسیم مناسب بین زیرسیستم‌های قابل و غیر قابل مشاهده را انجام دهد.

مراجع

- https://arxiv.org/pdf/2109.11919
- https://arxiv.org/pdf/2109.11919
- https://arxiv.org/pdf/2109.11919
- https://chatgpt.com/