

## «به نام خداوند جان و خرد»



دانشگاه صنعتی خواجه نصیرالدین طوسی

نام و نام خانوادگی: مبینا یوسفی مقدم

شماره دانشجویی: ۴۰۱۲۴۰۹۳

موضوع: گزارش پژوهش پایانی

درس سیستم‌های کنترل خطی

نام استاد: جناب آقای دکتر تقی‌راد

۱۴۰۳ بهمن ماه

## چکیده

در این پژوهش، داده‌های فرکانسی یک سیستم خطی تحلیل شده‌اند. ابتدا نمودارهای دامنه و فاز سیستم رسم شده‌اند، سپس یکتابع تبدیل برای سیستم تخمین زده شده است. در نهایت، پاسخ فرکانسی سیستم با استفاده از نمودار بود و پایداری سیستم با استفاده از مکان ریشه‌ها تحلیل شده‌اند.

هدف این پژوهش تحلیل و مدل‌سازی یک سیستم کنترل خطی با استفاده از داده‌های فرکانسی است. داده‌های فرکانسی شامل مقادیر اندازه‌گیری شده‌ی دامنه (Phase) و فاز (Magnitude) در فرکانس‌های مختلف هستند. این داده‌ها با استفاده از نرم‌افزار MATLAB بارگذاری شده‌اند و سپس به یکتابع تبدیل (Transfer Function) تبدیل شده‌اند. در ادامه، پاسخ فرکانسی و مکان ریشه‌های سیستم تحلیل شده‌اند.

### روش‌شناسی سوال (۱)

#### سوال اول

در فایل [Data.mat](#) مشخصات پاسخ فرکانسی یک سیستم موجود است. این دیتا شامل اندازه فاز و فرکانس سیستم است. در اولین مرحله، با استفاده داده‌های موجود، دیاگرام بودی سیستم را رسم کنید.

در این سوال، در تلاشیم با استفاده از نرم‌افزار MATLAB ابتدا اطلاعات داده شده را استخراج کنیم و دامنه و فاز را رسم کنیم. همچنین نهایتاً سعی می‌کنیم از ایجاد error های احتمالی در صورت خالی بودن فایل داده جلوگیری کنیم تا در ادامه در صورت بروز مشکل راحت‌تر بتوان آن را رفع کرد.

```
data = load('Data.mat');
magnitude = data.Data.magnitude;
phase = data.Data.phase;
frequency = data.Data.omega;
```

شکل ۱-۱

در مرحله اول فایل data.mat را دانلود کرده و در آدرس مربوط به پروژه قرار می‌دهیم تا در استخراج داده و ذخیره آن در متغیر های magnitude, phase & frequency مشکلی ایجاد نشود. سپس این اطلاعات را در متغیرهای یاد شده ذخیره می‌کنیم تا به توانیم نمودارهای خواسته شده را رسم کنیم.

```
figure;
subplot(2, 1, 1); %to ye safhe data nemoodar--->nemoodar aval
semilogx(frequency, 20*log10(magnitude)); %mehvar frequency logaritmi
xlabel('Frequency (rad/s)');
ylabel('Magnitude (dB)');
title('Magnitude');
grid on;

subplot(2, 1, 2); %nemoodar doyom hamoon safhe
semilogx(frequency, phase);
xlabel('Frequency (rad/s)');
ylabel('Phase (deg)');
title('Phase');
grid on;
```

شکل ۲-۱

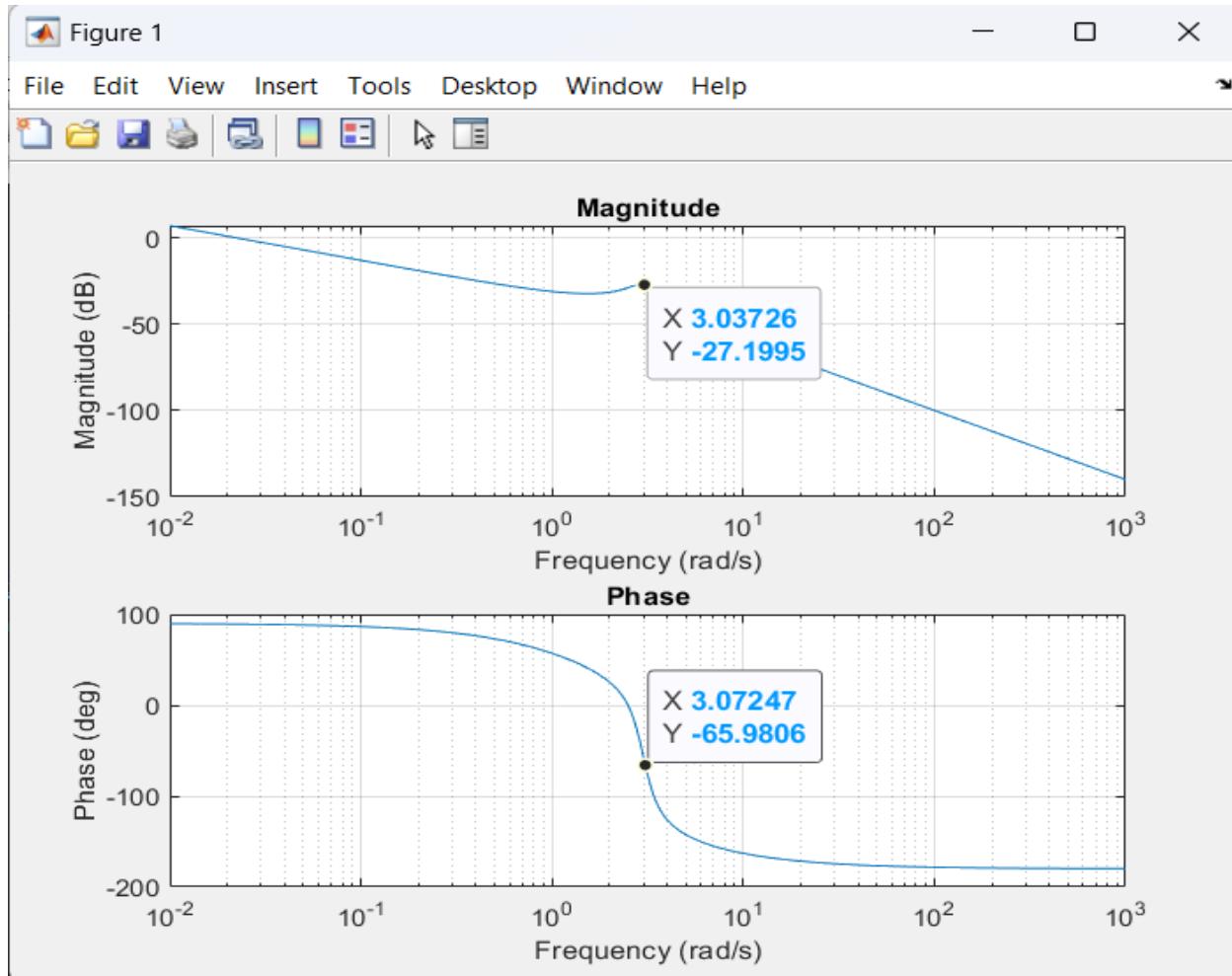
هدف این بخش که رسم نمودار دامنه و فاز است. با استفاده از subplot می‌توان در یکصفحه نمودارها را نمایش داد که تحلیل و مقایسه را سهولت می‌بخشد. همچنین با رسم نمودار دامنه و فاز در اسکیل لگاریتمی می‌توان محتوایه بیشتری از داده را در یکصفحه نمایش داد. این به معنی شناخت بهتر سیستم به دلیل اطلاعات بیشتر از آن است. به این منظور ازتابع semilogx استفاده می‌کنیم تا

محور افقی لگاریتمی باشد. توجه به فرمایید که در رسم دامنه، مقدار آن به db تبدیل شده است. با استفاده از xlabel & ylabel و نامگذاری محور ها و همچنین استفاده از title برای هر نمودار می توانیم دامنه و فاز را از هم تمیز دهیم.

```
if ~isfield(data, 'Data') || ~isfield(data.Data, 'magnitude') || ~isfield(data.Data, 'phase') || ~isfield(data.Data, 'omega')
    error('Required data are missing in the data file or data file is completely empty');
end
% dar soorati ke file data.mat dakhelesh on etelaat nabasho khabar mide ke
% file moshkel dare
```

شکل ۱-۳

همان طور که قبلاً اشاره شد این بخش کد به جهت رفع ساده تر مشکل در صورت نبودن داده ها مورد نیاز در فایل data.mat نوشته شده است. برای این بخش از chatgpt کمک گرفته ایم.



شکل ۱-۴

در شکل ۱-۴ خروجی مربوط به سوال ۱ مشاهده می شود که شامل دامنه و فاز است. هر دو نمودار همان طور که پیشتر توضیح داده شد بر مبنای لگاریتمی رسم شده اند. دامنه بر حسب db و فاز بر حسب درجه است. همانطور که می بینید در فرکانس ۳ رادیان بر ثانیه دامنه مقداری افزایش داشته این به دلیل حضور صفر غیر کمینه فاز است. هرچه زتا یا همان damping ratio کاهش یابد مقدار این چهش بیشتر می شود و در صورت تحریک سیستمی با زتا کوچک در فرکانسی که این چهش رخ می دهد، امکان بروز خطرات جدی وجود دارد. برای مثال در هنگام وقوع زلزله اگر فرکانس زلزله با این فرکانس برابر یا نزدیک به آن باشد خرابی های جبران ناپذیری به بار می آورد.

## روش شناسی سوال (۲)

سوال دوم

با استفاده از پاسخ فرکانسی داده شده، موارد زیر را بیابید:

• نوع سیستم

• مرتبه سیستم

• میزان تأخیر سیستم

• کمینه فاز بودن سیستم

قبل از پرداختن به حل سوال، ضروری است که ابتدا نمودار اندازه و فاز را به دقت تحلیل کنیم تا بتوانیم به هر بخش پاسخ دقیقی بدهیم. همانطور که از نمودار اندازه پیداست، سیستم در فرکانس‌های پایین با شبیب منفی  $20$  دسی‌بل آغاز می‌شود. در نمودار فاز نیز مشاهده می‌شود که فاز از مقدار مثبت  $90$  درجه شروع می‌کند، در حالی که باید از منفی  $90$  درجه آغاز می‌شد. این تغییر فاز به اندازه  $180$  درجه نشان‌دهنده تأثیر یک تغییر فاز  $180$  درجه‌ای در سیستم است.

در ادامه، اگر به نمودار اندازه توجه کنیم، در فرکانس  $2$  رادیان بر ثانیه مشاهده می‌کنیم که نمودار تقریباً به ثبات می‌رسد و افزایش  $20$  دسی‌بل در سیستم نمایان می‌شود. این تغییرات نشان‌دهنده وجود یک صفر در سیستم در این فرکانس است. در فرکانس  $3$  رادیان بر ثانیه، افزایش ناگهانی در اندازه دیده می‌شود و همزمان تغییرات قابل توجهی در فاز اتفاق می‌افتد. در این نقطه، فاز به مقدار  $180$  درجه کاهش می‌یابد که به وضوح نشان‌دهنده حضور یک قطب مرتبه دوم در این فرکانس است. از سوی دیگر، کاهش فاز  $90$  درجه‌ای نیز به علت وجود یک صفر غیرکمینه فاز در سیستم است.

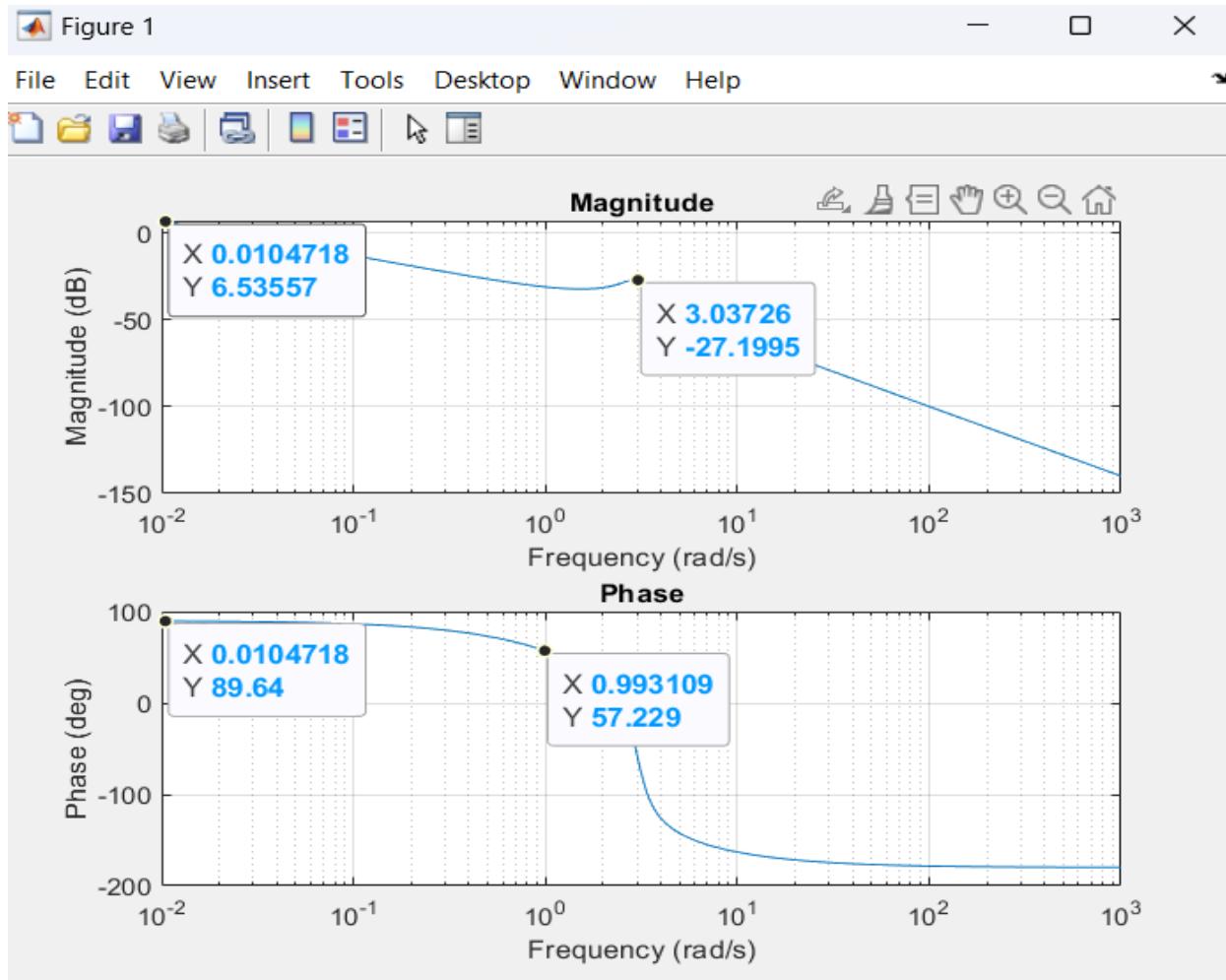
در ناحیه فرکانس‌های بالا، مشاهده می‌کنیم که نمودار اندازه با شبیب منفی  $40$  دسی‌بل ادامه می‌یابد. این وضعیت نشان‌دهنده آن است که تعداد قطب‌ها  $2$  تا بیشتر از صفر هاست. بنابراین، وجود یک صفر و سه قطب در سیستم منطقی به نظر می‌رسد. فرکانس‌های مربوط به صفرها و قطب‌ها نیز از نمودار به وضوح قابل شناسایی است.

در خصوص نوع سیستم، از آنجا که نمودار اندازه با شبیب منفی  $20$  دسی‌بل شروع می‌شود، می‌توان نتیجه گرفت که سیستم از نوع تیپ یک است و بیانگر وجود یک قطب در مبدأ است.

مرتبه سیستم از روی درجه مخرج تابع تبدیل آن مشخص می‌شود. همان‌طور که پیشتر اشاره شد، این تابع تبدیل دارای سه قطب است و بنابراین، مرتبه سیستم برابر با  $3$  خواهد بود.

تاخیر سیستم معمولاً از طریق تغییرات فاز نسبت به تغییرات فرکانس در نمودار نشان داده می‌شود. هرچه این تغییرات فاز بیشتر و شدیدتر باشد، تاخیر سیستم افزایش می‌یابد، و هرچه تغییرات فاز کمتر باشد، تاخیر سیستم کاهش پیدا می‌کند. برای محاسبه تقریبی این تاخیر، ابتدا باید شبیب خط وصل‌کننده دو نقطه از نمودار را محاسبه کنیم. برای این منظور، مقدار فاز را در فرکانس  $10^4$  که تقریباً می‌توان آن را برابر با صفر در نظر گرفت، خوانده و سپس همین کار را برای فرکانس  $0.09$  که تقریباً همان  $1$  رادیان بر ثانیه است نیز انجام می‌دهیم(شکل ۱-۲). پس از این مرحله، شبیب این خط را محاسبه می‌کنیم. نکته‌ای که باید به آن توجه کنیم این است که باید زاویه‌ها را از درجه به رادیان تبدیل کنیم. برای این تبدیل، از فرمول  $2\pi/360$  استفاده می‌کنیم و سپس تاخیر سیستم را بر اساس این تغییرات فاز محاسبه خواهیم کرد. نتیجه این محاسبات نشان می‌دهد که تقریباً نیم ثانیه تاخیر در سیستم وجود دارد.

این سیستم غیرکمینه فاز است. همان‌طور که پیشتر اشاره شد، در فرکانس  $2$  رادیان بر ثانیه، نمودار اندازه به طور تقریبی ثابت می‌شود که نشان‌دهنده افزایشی  $20$  دسی‌بل به شبیب منفی  $20$  دسی‌بل است. این تغییر باعث می‌شود نمودار تقریباً ثابت بماند. همچنین، در نمودار فاز مشاهده می‌شود که فاز به تدریج کاهش می‌یابد. این کاهش فاز نشان‌دهنده وجود یک صفر غیرکمینه فاز در سیستم است، صفرهایی که باعث کاهش مقدار فاز در این فرکانس‌ها می‌شوند.



شکل ۱-۲

در ادامه محاسبات توضیحات فوق ارائه خواهد شد:

$$T_{\text{delay}} = \frac{d\phi}{dw}$$

$$w = 0 \rightarrow \phi = 89.64$$

$$w = 1 \rightarrow \phi = 57.229$$

توجه فرمایید فرکانی های فوق تقریبی هستند.

$$T_{\text{delay}} = ((57.229 - 89.64)(2\pi/360))/(0-1) = 0.5656 \text{ s}$$

این سیستم تقریبا با نیم ثانیه تاخیر واکنش نشان می دهد.

وجود صفر غیرکمینه فاز در سیستم باعث می شود که نمودار فاز بهطور پیوسته کاهش یابد، که این امر خود یک ویژگی بارز در سیستم های غیرکمینه فاز بهشمار می رود. از سوی دیگر، وجود چنین صفری در سیستم موجب می شود که سیستم در پاسخ فرکانسی خود تغییرات فازی شدیدتری را تجربه کند، که به نوبه خود می تواند بر عملکرد و تاخیر سیستم تأثیر بگذارد.

### روش شناسی سوال (۳)

با توجه به پاسخ فرکانسی داده شده، به سیستم یکتابع تبدیل مناسب برآذش کنید. باید از مدل بدست آمده در این بخش، در ادامه برای کنترل سیستم استفاده نمایید.

با استفاده از ابزار SystemIdentification نرم افزار متلب مدل مناسبی برآذش کنید.

برای حل این سوال از اطلاعات سوال قبلی استفاده شده و ابتدا برای آن حل دستی انجام می‌دهیم.

در ابتدا، نمودار  $\frac{db}{decade}$ - $20$ - $\omega$  می‌باشد. می‌توان قطب در مبدأ درجه دارم. در فرکانس

$2 \frac{rad/sec}{decade}$  نازیم کاصلت بیتی داشته ریب بانده  $20 \frac{db}{decade}$  را با فرآیند بنابراین می-

حدن غیرلینی ناز در این فرکانس خارج. همچنان در  $3 \frac{rad/sec}{decade}$  تغییر ناچاری زیاد شدن نزدیکی را داریم و نمودار

نوزنی در این فرکانس به قدر ناچاری - مقدار زیادی کاصلت پیدا کرده است. پس یک قطب درجه ۲ درجه

$$\omega_1 = 2 \frac{rad/s}{s} \rightarrow z_1 = 0.5 \quad ; \quad \omega_2 = 3 \frac{rad/s}{s} \rightarrow z_2 = \frac{1}{3}$$

مقدار  $z_2$  را نیز به حس و ذهن می‌دانیم و مقدار بیتی نزدیکی خارج باشد بسیار زیادی کاصلت

$\xi = 0.15$  پیدا کرد. مخفف است که  $\xi$  حبابی نزدیکی حفظ بوده.

$$L(s) = \frac{1 - 0.5s}{s(1, s^2 + 2/3 \times 0.15s + 1)} = \frac{1 - 0.5s}{s(1, s^2 + 0.1s + 1)}$$

حال با این  $\xi$  آن افرازه سرور تابع تحریک خود را با لشیه نویز و مطابق بودی سینه شود.

با فرب مقدار مخفف مخفف  $\xi$  باید  $0.1$  درآی خوب شد:

$$L(s) = 0.1 \times \frac{1 - 0.5s}{s(1, s^2 + 0.1s + 1)}$$

شکل ۱-۳

در شکل ۱-۳ تابع تبدیل با توجه به نمودار بودی سیستم داده شده تخمین زده شده است.

در سوال ۱-۳ از ما خواسته شده که با استفاده از ابزار System Identification، سیستم مورد نظر را شبیه‌سازی و تابع تبدیل آن را تخمین بزنیم. برای انجام این کار، از توابع موجود در System Identification Toolbox استفاده کردیم تا سیستم را تقریب بزنیم. یکی از روش‌های انجام این کار این است که ابزار Command در System Identification باز کرده و از آنجا با تعیین تعداد صفرها و قطبها، همچنین مشخص کردن پیوسته یا گستته بودن سیستم، تابع Window تبدیل سیستم را تخمین بزنیم. پس از تخمین تابع تبدیل، می‌توانیم آن را از workspace استخراج کرده و در ادامه از آن برای تحلیل‌های بیشتر استفاده کنیم.

در MATLAB، یکی از توابع کلیدی برای این کار، تابع `tfest` است که برای تخمین مدل‌های تابع تبدیل استفاده می‌شود. با این تابع می‌توان تعداد صفرها و قطبها را به طور دستی تعیین کرده و تابع تبدیل سیستم را بر اساس داده‌های ورودی- خروجی شبیه‌سازی کرد. همچنین، با استفاده از تابع `iddata` می‌توان داده‌های ورودی و خروجی سیستم را به فرمت مناسب تبدیل کرد تا در فرآیند شناسایی سیستم استفاده شود.

روش دیگر، استفاده از کدهای MATLAB و توابع موجود در این `toolbox` به صورت مستقیم است. برای این کار، ابتدا باید داده‌های ورودی و خروجی سیستم را جمع‌آوری کرده و از تابع `ident` برای بارگذاری داده‌ها و شروع فرآیند شناسایی استفاده کنیم. سپس با استفاده از تابع `arx` یا `tfest` می‌توانیم مدل‌های مختلفی از سیستم را تخمین بزنیم.

برای انجام تخمین دقیق‌تر، می‌توان از روش‌های اعتبارسنجی و ارزیابی کیفیت مدل استفاده کرد که در System Identification Toolbox ابزارهایی همچون `compare` وجود دارد. این ابزار امکان مقایسه خروجی مدل شبیه‌سازی شده با داده‌های واقعی را فراهم می‌کند و به ما کمک می‌کند تا میزان دقت مدل تخمین‌زدشده را ارزیابی کنیم.

در نهایت، استفاده از توابع موجود در این `toolbox` به ویژه در تحلیل‌های پیچیده، باعث می‌شود تا فرآیند شبیه‌سازی و شناسایی سیستم‌ها سریع‌تر و دقیق‌تر انجام شود و نیاز به تحلیل دستی کاهش یابد.

```

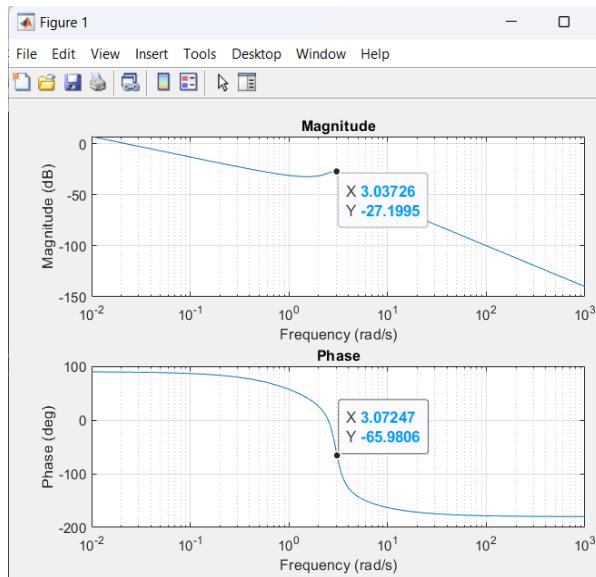
1 data = load('Data.mat');
2 magnitude = data.Data.magnitude;
3 phase = data.Data.phase;
4 frequency = data.Data.omega;
5
6 ghotbi = magnitude .* exp(1j * deg2rad(phase)); %form ghotbi
7
8
9 freqresdata = idfrd(ghotbi, frequency, 0); % Ts = 0 tebgh chat gpt chon system continuous bood ts=0 va agar discrete mishod ts>0
10 %system identification
11 %pasokh frequency misaze az dade ha
12 esystf = tfest(freqresdata, 3, 1);% in function ba tavajoh be system identification va tedad sefr va ghotb ha yek transfer function misazad
13 disp('Estimated Transfer Function:');
14 disp(esystf);
15
16
17 figure;
18 bode(esystf); %in nemoodar bode transfer function tarahi shodas
19 xlim([10e-4, 10e3]);
20 grid;
21 title('Bode Diagram of Estimated Transfer Function');
22
23 if ~isfield(data, 'Data') || ~isfield(data.Data, 'magnitude') || ~isfield(data.Data, 'phase') || ~isfield(data.Data, 'omega')
24 error('Required data are missing in the data file or data file is completely empty');
25 end
26 % dar soorati ke file data.mat dakhelesh on etelaat nabashe khabar mide ke
27 % file moshkel dare

```

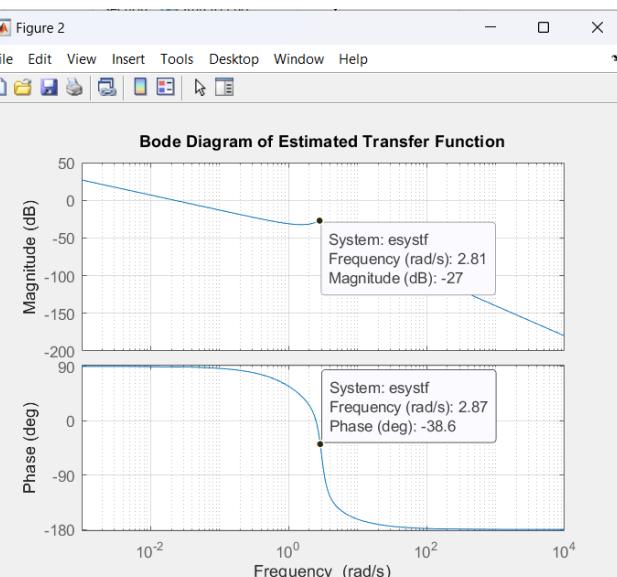
شکل ۲-۳

به جهت صرفه‌جویی در زمان بخش‌های تکراری که موجود در شکل ۲-۳ بررسی نخواهد شد. توضیحات را مشترکاً در روش شناسی سوال ۱ خدمتمندان داده‌ایم. برای این که بتوانیم پاسخ فرکانسی سیستم را با توجه به داده‌های مسئله به دست بیاوریم اینها را به فرم قطبی تبدیل می‌کنیم. این کار را در خط ۶ کرده‌ایم. سپس با استفاده از فرکانس، تعیین پیوستگی سیستم و فرم قطبی آن به عنوان ورودی `idfrd` می‌توانیم پاسخ فرکانسی آن را تخمین بزنیم. با استفاده از دستور `tfest` و پاسخ فرکانسی به دست آمده و تعداد صفر و قطب‌ها که در سوالات قبلی آنها را با تحلیل به دست آورده‌یم می‌توانیم تابع تبدیل سیستم را تخمین بزنیم. با مقایسه دامنه

و فاز سیستم تخمین زده شده با شکل ۴-۱ میبینیم که شباهت زیادی دارد و این به طرکی تابع تبدیل به دست آمده در شکل ۱-۳ و شکل ۲-۳ با تقریب خوبی یکسان است.



شکل ۴-۱



شکل ۳-۳

برای دیدن سیستم تخمین زده شده در خروجی کافی است نام آن (esystf) را صدا بزنیم. ادعای یکسان بودن تقریبی تابع تبدیل با مقایسه شکل زیر صحت بخشیده می‌شود. توجه کنید که معادله مشخصه دقیقاً یکسان و صفرهای سیستم‌های به دست آمده از دو روش مختلف بسیار به هم نزدیک هستند و با خطای بسیار کمی یکسان‌اند.

$$L(s) = 0.1 \times \frac{1 - 0.5s}{s(1, s^2 + 0.1s + 1)}$$

شکل ۵-۳

```
esystf =
  0.1 s - 0.2
  -----
  s^3 + 0.9 s^2 + 9 s
Continuous-time identified transfer function.

Parameterization:
  Number of poles: 3    Number of zeros: 1
```

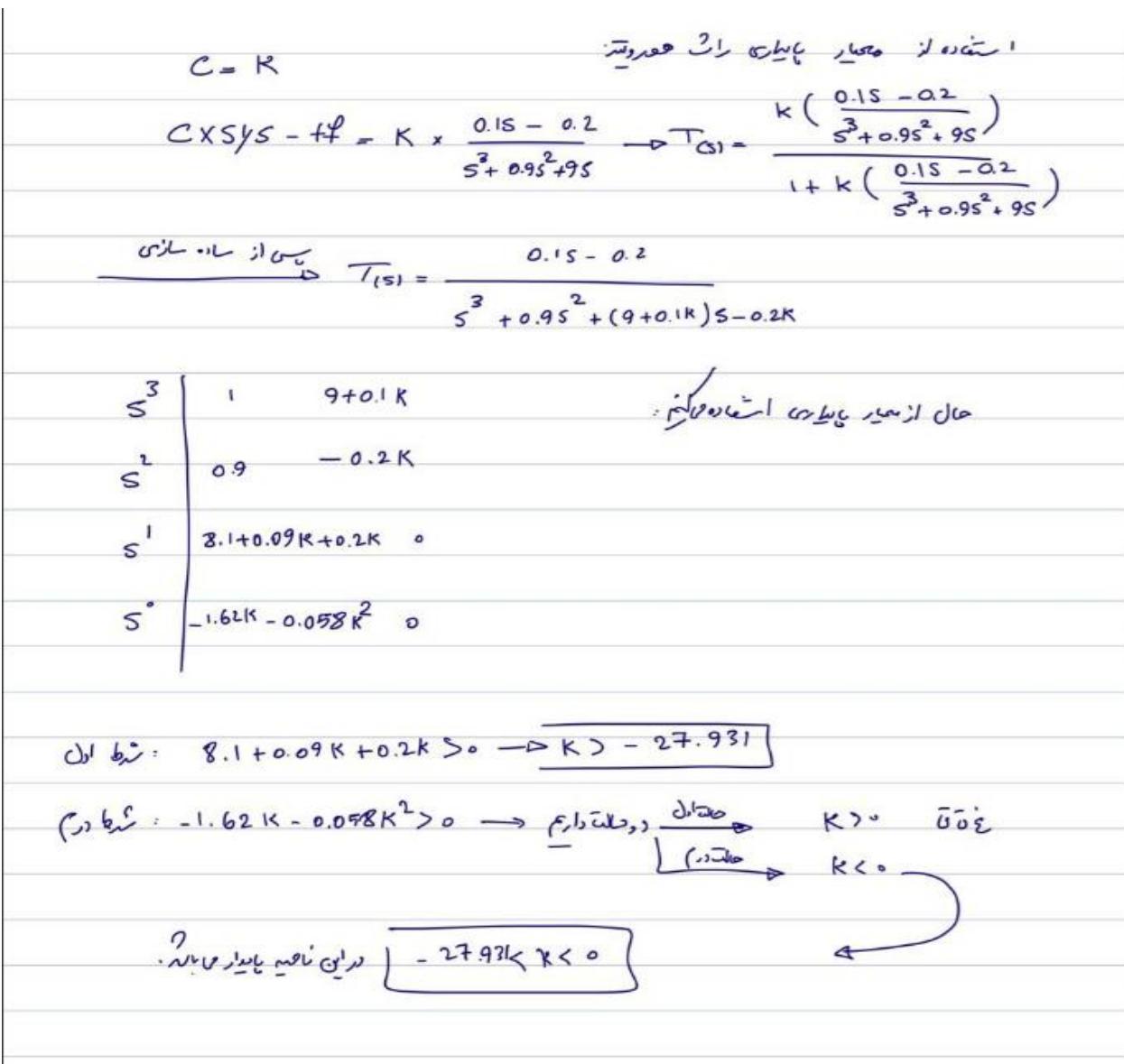
شکل ۴-۳

## روش شناسی سوال ۴)

با استفاده از معیار پایداری راث-هرویتز مشخص نمایید که به از ای چه مقدار بهره سیستم پایدار می‌ماند.

برای این کار ابتدا جدول مربوط به ضرایب را میکشیم و محاسبات لازم را طبق جزوه استاد انجام می‌دهیم. در نهایت شروط مورد نیاز جهت پایداری را اعمال می‌کنیم. بعضی از حالات غیر قابل قبول هستند که آنها در شکل ۴-۱ مشخص کرده‌اند. از اشتراک حالات قبل قبول ناجیه ای برای  $k$  به دست می‌آید که تمامی شروط را ارضاء می‌کند.

توجه کنید که معیار پایداری راث هرویتز فقط توانایی تحلیل پایداری را دارد.



شکل ۱-۴

### روش شناسی سوال ۵

مکان هندسی سیستم بدست آمده را رسم کنید. با استفاده از این نمودار تحلیل کنید که آیا می‌توان با استفاده از کنترل کننده تناوبی سیستم را به پایداری رساند؟ توضیح دهد.

با استفاده از کدام یک از کنترل کننده‌های صنعتی PI و PD می‌توانیم سیستم را به پایداری برسانیم؟ توضیح دهد.

در این سوال ابتدا به رسم‌مکان هندسی ریشه‌ها به ازای تمامی مقادی  $K$  می‌پردازیم. برای ای کار ازتابع تبدیل تخمین زده شده در سوال ۳ استفاده خواهیم کرد. در نهایت تأثیر کنترلر بر سیستم حلقه بسته جدید را با استفاده از نمودار بودی بررسی خواهیم کرد.

از آنجایی که دسترسی به اطلاعات فایل mat قبلا در روش شناسی سوال ۱ و همچنین تخمین سیستم از طریق system identification در روش شناسی سوال ۲ توضیح داده شده است، از تکرار این مطالب خودداری خواهم کرد.

در این کد برای رسم مکان هندسی به جای تابع rlocusplot از rlocus استفاده کرده‌ایم زیرا در ورژن matlab من grid روشی را روی این تابع جواب می‌داد. مکان هندسی ریشه‌ها به ازای  $K$  مثبت با خط ممتد و مکان هندسی ریشه‌ها با  $K$  منفی با نقطه خط کشیده شده است.

برای رسم مکان هندسی به ازای  $K$  منفی کافی است ورودی rlocusplot را به جای خود سیستم تخمین زده قرینه آن بدھیم.

توضیحات فوق مربوط به کد موجود در شکل ۱-۵ می‌باشد.

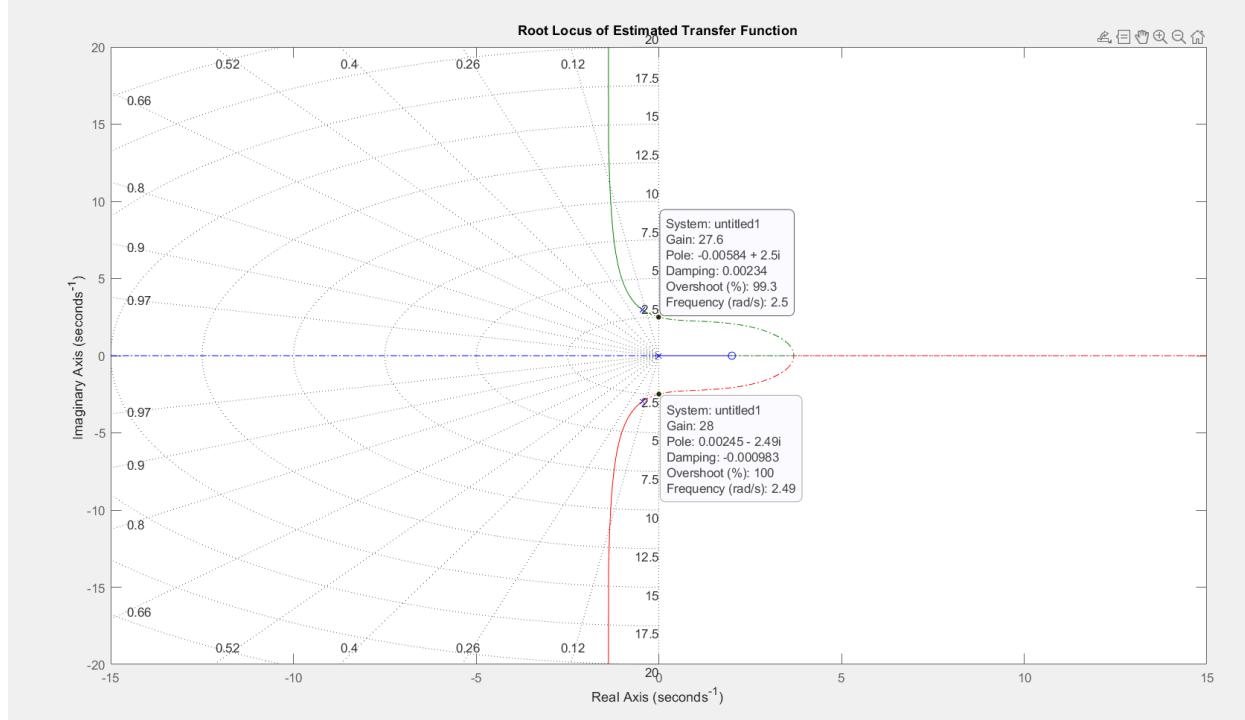
```

1   data = load('Data.mat');
2   magnitude = data.Data.magnitude;
3   phase = data.Data.phase;
4   frequency = data.Data.omega;
5
6
7   ghotbi = magnitude .* exp(1j * deg2rad(phase)); %form ghotbi
8
9
10 freqresdata = idfrd(ghotbi, frequency, 0); % Ts = 0 tebgh chat gpt chon system continuous bood ts=0 va agar discrete mishod ts>0
11 %system identification
12 %pasokh frequency misaze az dade ha
13 esystf = tfest(freqresdata, 3, 1); % in function ba tavajoh be system identification va tedad sefr va ghotb ha yek transfer function misazad
14 disp('Estimated Transfer Function:');
15 disp(esystf);
16
17
18
19 figure;
20 rlocusplot(esystf); %in nemoodar makan hendi rishe transfer function tarahi shodas
21 %grid in dorost kon
22 %in dastor rlocusplot ejaze grid mide rlocus gridesh faal
23 %nemishod
24 grid on;
25 title('Root Locus of Estimated Transfer Function');
26 hold on;
27 rlocusplot(-esystf, '-.'); %root locus be ezaye K<0 ham rasm kone
28 grid on;
29

```

شکل ۱-۵

در سوال ۴ با استفاده از روش راث هرویتز ناحیه پایداری را به دست آوردیم. طبق اطلاعات موجود در حل دستی آن سوال می‌توانیم نتیجه بگیریم که  $k$  باید منفی و بزرگتر از  $-27/93$  باشد. یا به عبارتی سیتم به ازای  $k = -27/93$  در مرز ناپایداری قرار دارد. این موضوع را مکان هندسی ریشه‌های سیستم تخمین زده شده نیز تایید می‌کند. در ادامه خروجی کد موجود در شکل ۱-۵ را خدمتتان ارائه می‌کنیم.



شکل ۲-۵

همانطور که قبل تر توضیح دادیم نقطه خط مکان هندسی ریشه ها به ازای  $k$  های منفی است. دو نقطه نمایش داده شده حول محور موهومی  $k$  منفی را نشان می‌دهند که به ازای آن ریشه ها در مرز ورود به OPRHP هستند و به سمت قطب های ناپایدار می‌روند. مفتخریم توجهتان را به داده های نمایش داده شده جلب کنیم، میبینید که با افزایش اندازه  $k$  میزان فراجهش به  $100\%$  یا همان نوسانی کامل افزایش می‌یابد و با ادامه این روند خروجی واگرا خواهد شد و سیستم ناپایدار می‌شود. در این فرآیند زتا یا همان damping ratio شدیداً کاهش می‌یابد و به صفر نزدیک می‌شود. اگر توضیحات سوال یک را به یاد داشته باشید خواهید دید که مقدار زتا کوچک در مواردی می‌تواند خرابی جبران ناپایداری ایجاد کند البته که این به معنای مطلقاً نامطلوب بودن این شرایط نیست، در سیاری از سیستم ها مانند amplifier های صوتی و سیستم هایی که سرعت زیادی دارند این خروجی ها می‌توانند ایده آل باشند. ما باید با توجه به سیستم و هزینه و محدودیت های فیزیکی مان مقدار trade off خود را بستنجیم. سیستم فوق به ازای تمامی  $k$  های مثبت یک قطب ناپایدار دارد در ادامه سعی می‌کنیم این سیستم را به جای فقط  $k$  با یک کنترلر پایدار کنیم.

با توجه به تحلیل فوق کنترلکننده ها حتما باید دارای  $k$  منفی باشند تا سیستم پایدار بماند. با توجه به تست های sisotool میبینیم که کنترل کننده pd این سیستم را ناپایدار خواهد کرد و به همین دلیل از کنترلر pi استفاده میکنیم مشکلی در پایداری سیستم ایجاد نمیکند.

```

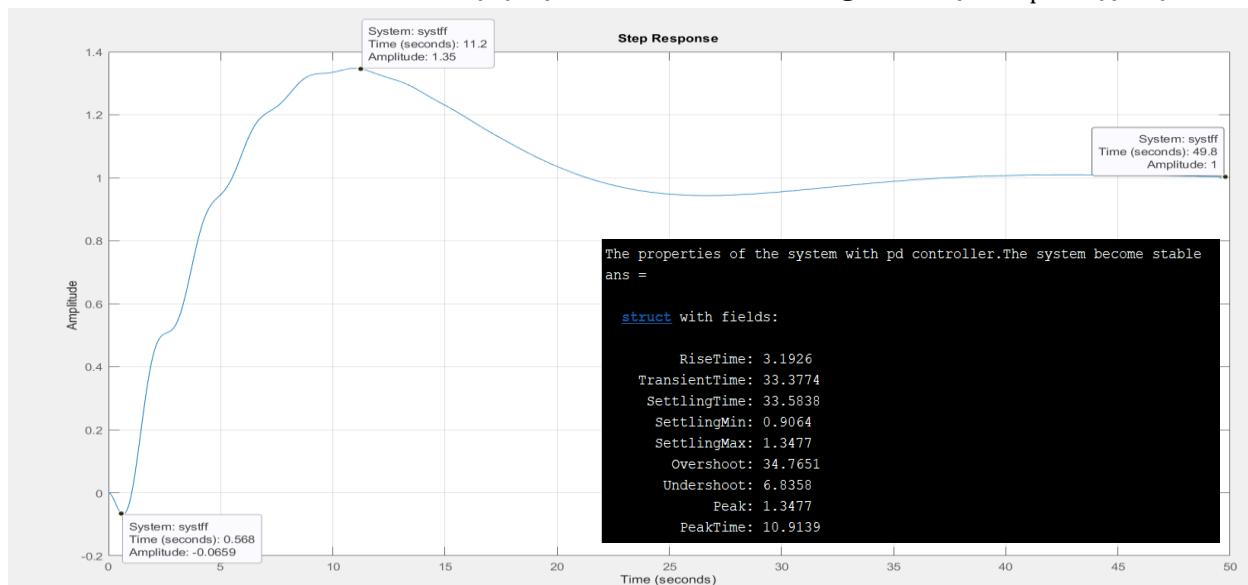
1  data = load('Data.mat');
2  magnitude = data.Data.magnitude;
3  phase = data.Data.phase;
4  frequency = data.Data.omega;
5
6
7  ghotbi = magnitude .* exp(1j * deg2rad(phase)); %form ghotbi
8
9
10 freqresdata = idfrd(ghotbi, frequency, 0); % Ts = 0 tebgh chat gpt chon system continuous bood ts=0 va agar discrete mishod ts>0
11 %system identification
12 %pasokh frequency misaze az dade ha
13 esystf = tfest(freqresdata, 3, 1); % in function ba tavajoh be system identification va tedad sefr va ghotb ha yek transfer function misazad
14 disp('Estimated Transfer Function:');
15 disp(esystf);
16
17
18 figure;
19 rlocusplot(esystf); %in nemoodar makan hendesi rishe transfer function tarehi shodas
20 %grid in dorost kon
21 %in dastor rlocusplot ejaze grid mide rlocus gridesh faal
22 %nemishod
23
24 grid on;
25 title('Root Locus of Estimated Transfer Function');
26 hold on;
27 rlocusplot(-esystf, '-.'); %root locus be ezaye K<0 ham rasm kone
28 grid on;
29
30 s=tf('s');
31 time=0:0.0001:50;
32 controller=-10*(s+0.2)/s;
33 loopgain=controller*esystf;
34 systff=loopgain/(1+loopgain);
35 figure;
36 step(systff,time);
37 grid;
38 fprintf("The properties of the system with pd controller.The system become stable");
39 stepinfo(systff)
40
41 if ~isfield(data, 'Data') || ~isfield(data.Data, 'magnitude') || ~isfield(data.Data, 'phase') || ~isfield(data.Data, 'omega')
42 error('Required data are missing in the data file or data file is completely empty');
43 end
44 % dar soorati ke file data.mat dakhelesh on etelaat nabashe khabar mide ke
45 % file moshkel dare

```

شکل ۳-۵

در شکل فوق توجه شمارا به کد های موجود در خط ۳۰ الی ۳۹ جلب میکنیم. ایندا تابع تبدیل کنترلگر را تعریف میکنیم سپس آن را در مسیر پیشرو قرار میدهیم و تابع تبدیل جدید را مینویسیم . سپس با استفاده از دستور step میتوانیم پاسخ پله سیستم جدید را در بازه زمانی مشخص شده نمایش دهیم.

با استفاده از دستور stepinfo میتوانید اطلاعاتی مثل overshoot، rise time و غیره را ببینید.



شکل ۴-۵

با توجه به شکل ۴-۵ این سیستم دارای حدود ۳ ثانیه rise time است و بعد از ۳۳ ثانیه تقریباً خطای ماندگار آن به ورودی پله صفر می‌شود، با حدود ۶٪ فروجehش با توجه به همگرایی آن میتوان گفت که سیستم با کنترلگر pd پایدار است.

## روش شناسی سوال ۶

در سیستمی که بدست آورده اید، عبارت  $s-a/s$  را از مدل حذف کنید. یعنی سیستم هیچی صفر و هیچ قطبی در مبدأ نداشته باشد. سپس برای این سیستم کنترل کننده ای طراحی کنید که فراجهشی بین ۰-۱۵ درصد داشته باشد و زمان نشست هم کمتر از ۰-۱ ثانیه باشد. پس از طراحی این کنترل کننده، خروجی سیستم از رسم کرده و با سیستم کنترل نشده مقایسه کنید.

برای حل این سوال، دو روش مختلف را امتحان کردیم ابتدا با استفاده از ابزار SISOTool در MATLAB و نهایتاً با تحلیل دستی تحلیل کردیم. در روش اول، هدف ما این بود که هم زمان نشست سیستم را بهینه کنیم و هم فراجهش آن را کاهش دهیم. برای رسیدن به این هدف، از دو کنترلر Lead استفاده کردیم. دلیل این انتخاب این است که کنترلر های Lead می‌توانند دینامیک سیستم را افزایش دهند و در نتیجه باعث بهبود سرعت پاسخ سیستم شوند، که این تغییر سرعت به کاهش زمان نشست کمک می‌کند. همچنین این کنترلر توансه فراجهش سیستم را به خوبی تنظیم کند. دلیل این موضوع این است که کنترلر Lead با ایجاد صفرهای غالب و قطب‌های دور، باعث می‌شود تا مکان هندسی ریشه‌های سیستم به سمت چپ جابجا شوند. این جابجایی مکان هندسی ریشه‌ها باعث می‌شود که فراجهش سیستم کاهش یابد و پاسخ سیستم پایدارتر شود.

برای تنظیم پاسخ سیستم، گین حدودی در حدود ۷۰۰ انتخاب شد. این مقدار گین به این دلیل انتخاب شد که خود سیستم اولیه نسبتاً لخت و ضعیف بود و نمودار بودی آن در پایین تر از ۰ دسی‌بل قرار داشت لذا نیاز مند به کنترلری با گین بالا می‌باشد تا بتواند در ولکشن به ورودی پله به شرایط مطلوب نزدیک شود و به مقدار یک برسد، توجه نمایید این افزایش گین با دقت زیاد انجام شد، چرا که با توجه به محدودیت‌های کنترلر، نمی‌توانست گین را بیش از حد بالا ببرم. و به عبارتی control effort خیلی بالا می‌رفت و پیاده سازی آن به دلیل محدودیت‌های فیزیکی غیر ممکن می‌شد.

در نهایت، با این مقدار گین، توانستیم پاسخ مناسبی از سیستم بگیریم که در نمودارهای خروجی قابل مشاهده است. کدها و نتایج مربوط به این طراحی بهطور کامل در ادامه آورده شده است.

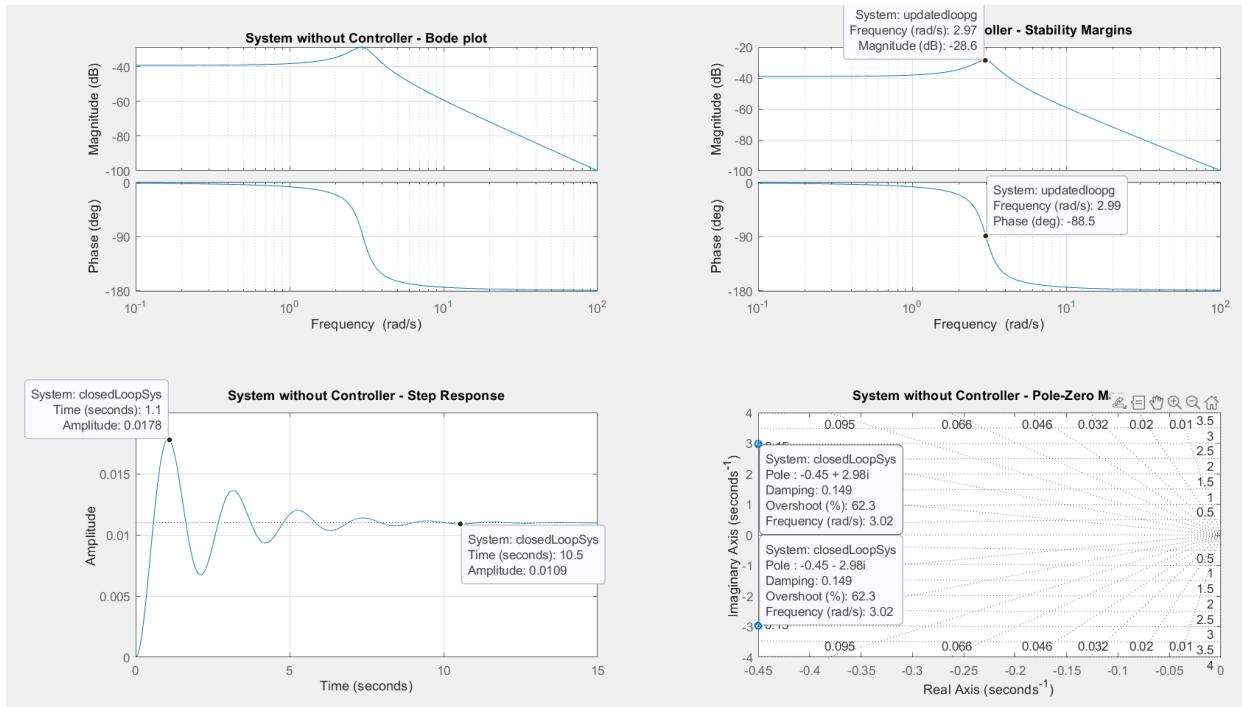
```

1  data = load('Data.mat');
2  magnitude = data.Data.magnitude;
3  phase = data.Data.phase;
4  frequency = data.Data.omega;
5
6  % Inja transfer function system ro bedune controller taarif mikonim
7  s = tf('s');
8  updatedloopg = 0.1 / (s^2 + 0.9*s + 9);
9
10 % Vaqte simulation ro baraye tahlil mizanimo ke ye noghte shoroo payan dare
11 % va ye vectore
12 t = 0:0.001:15;
13
14 %system ro bedune controller tahlil mikonim
15 % Aval Bode plot ro mizarm, ta bebinim chera system chetor kar mikone
16 figure;
17 subplot(2, 2, 1);
18 bode(updatedloopg);
19 grid on;
20 title('System without Controller - Bode plot');
21
22 % Stability margins ro mizarem, ta bebinim system cheghadr stable
23 subplot(2, 2, 2);
24 margin(updatedloopg);
25 grid on;
26 title('System without Controller - Stability Margins');
27
28 % Hala step response ro mizarm
29 closedLoopSys = updatedloopg / (1 + updatedloopg);
30 subplot(2, 2, 3);
31 step(closedLoopSys, t);
32 grid on;
33 title('System without Controller - Step Response');
34 stepInfo = stepinfo(closedLoopSys);
35 disp('System without Controller - Step Response Info:');
36 disp(stepInfo);
37
38 % Pole-zero map ro mizarem, ke bebinim ghotb v sefr napaydar kojan va
39 % betonim ba controller system paydar kardi
40 subplot(2, 2, 4);
41 pzmap(closedLoopSys);
42 grid on;
43 title('System without Controller - Pole-Zero Map');
44

```

شکل ۶-۱

کد فوق به جهت مقایسه بهتر تأثیر اضافه شدن کنترلگر به سیستم نوشته شده است. به وسیله این کد می‌توان دامنه و فاز سیستم حلقه باز بدون کنترلگر و پاسخ پله و محل قرار گیری صفر و قطب های سیستم حلقه بسته را مشاهده کرد.



شکل ۲-۶

مانظور که در شکل فوق مشهود است سیستم، نمودار بودی اول مانند سوال های قبلی حضور یک صفر غیر کمینه فاز را نشان می‌دهد. در نمودار دومی میبینید که سیستم دارای نوسانات است و خطای ماندگار آن به ورودی پله بسیار زیاد است. این به دلیل همان لختی سیستم است که قبل از این را مطرح کردیم. همچنین میتوانید قطب های سیستم را نیز مشاهده کنید که در مرحله بعد با اضافه شدن کنترلر تغییراتی در این صفر و قطب هارخ می‌دهد. در ادامه میتوانید اطلاعات اطلاعات مربوط به پاسخ پله قبل از اضافه کردن کنترلگر را نیز مشاهده کنید.

```
System without Controller - Step Response Info:
    RiseTime: 0.3895
    TransientTime: 8.5866
    SettlingTime: 8.5866
    SettlingMin: 0.0068
    SettlingMax: 0.0178
    Overshoot: 61.9998
    Undershoot: 0
    Peak: 0.0178
    PeakTime: 1.0234
```

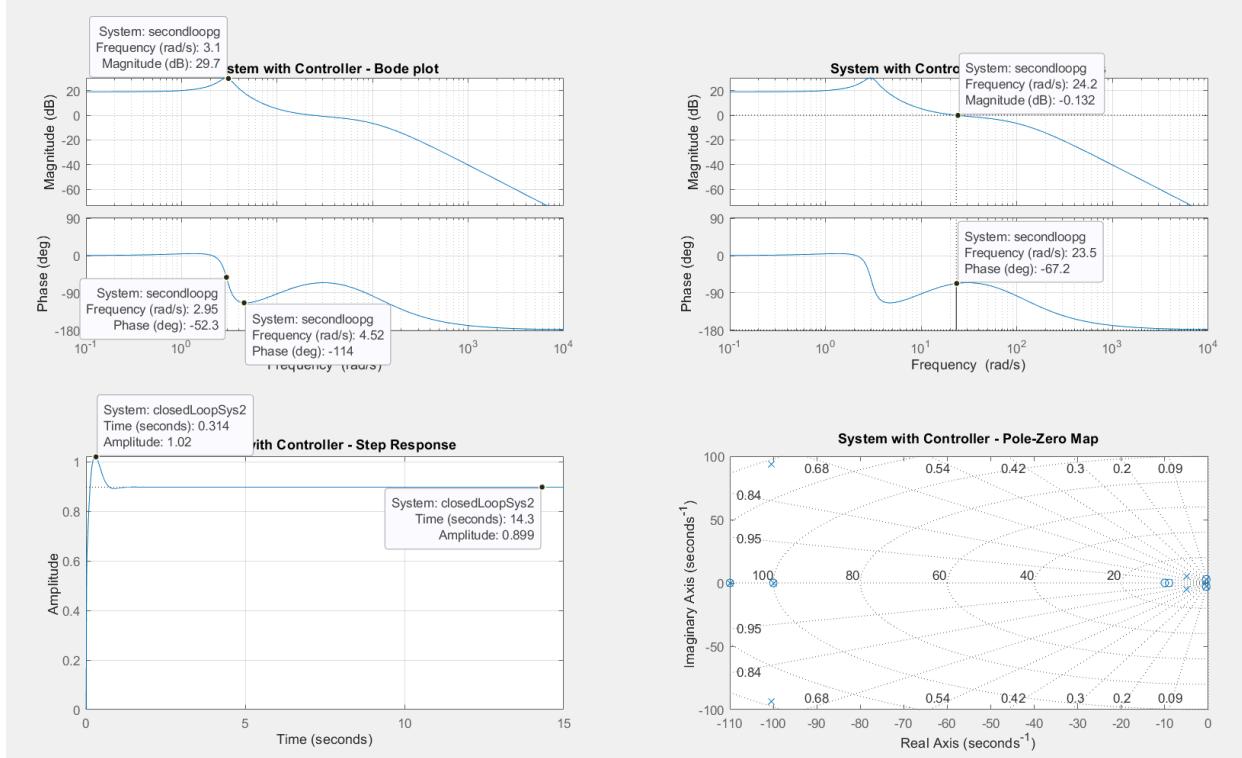
شکل ۳-۶

حال کنترلگر را در مسیر پیش رو قرار می‌دهیم وتابع تبدیل سیستم حلقه بسته را به دست می‌آوریم. نتایج حاصل را با شکل ۲-۶ مقایسه خواهیم کرد.

```
45 % Hala controller dovom ro tarif mikonim, chon mikayem ba controller bebinim system chi mishe
46 secondcontroller = 97380 * ((9 + s) * (10 + s)) / (((110 + s) * (100 + s));
47 secondloopg = secondcontroller * updatedloopg;
48
49 % Hala system ro ba controller tahlil mikonim
50 % Ayal Bode plot ro mizarem baraye system ba controller
51 figure(1);
52 subplot(2, 2, 1);
53 bode(secondloopg);
54 grid on;
55 title('System without Controller - Bode plot');
56
57 % stability margins ba controller ro mizarem
58 subplot(2, 2, 2);
59 margin(secondloopg);
60 grid on;
61 title('System without Controller - Stability Margins');
62
63 % Hala step response ba controller ro mizarem
64 closedLoopsys2 = secondloopg / (1 + secondloopg);
65 subplot(2, 2, 3);
66 step(closedLoopsys2, t);
67 grid on;
68 title('System without Controller - Step Response');
69 stepInfo2 = stepinfo(closedLoopsys2);
70 disp('System without Controller - Step Response Info:');
71 disp(stepInfo2);
72
73 % Pole-zero map ba controller ro mizarem
74 subplot(2, 2, 4);
75 pzmap(closedLoopsys2);
76 grid on;
77 title('System without Controller - Pole-Zero Map');
```

شکل ۴-۶

کد موجود در شکل ۴-۶ برای مقایسه نمودار های حلقه بازو بسته بدون کنترلر و با کنترلر نوشته شده است بدینهی است که در استفاده توابع تغییری رخ نداده و تنها ورودی های آنها تغییر کرده اند.



شکل ۴-۶

در نمودار اول میبینید که دامنه به اندازه  $60^\circ$  دسیبل در فرکانس های افزایش یافته و این موضوع به خودی خود باعث کاهش خطای ماندگار بر اساس قضیه مقدار نهایی خواهد شد. این موضوع را میتوان در نمودار دوم مشاهده کرد که خطای ماندگار به ورودی پله واحد از  $0/0.99$  به کاهش یافته است. همچنین با مقایسه دو شکل ۴-۵ و ۴-۶ متوجه کاهش نوسانات میشویم که ما را به پایداری نزدیک تر نیز میکند. با بالا امدن نمودار دامنه این نمودار به مقدار  $0^\circ$  دسیبل نیز میرسد این به معنی توانایی محاسبه حاشیه فاز میتوانیم پایداری این سیستم را بررسی کنیم همچنان حاشیه بهره بینهایت است.

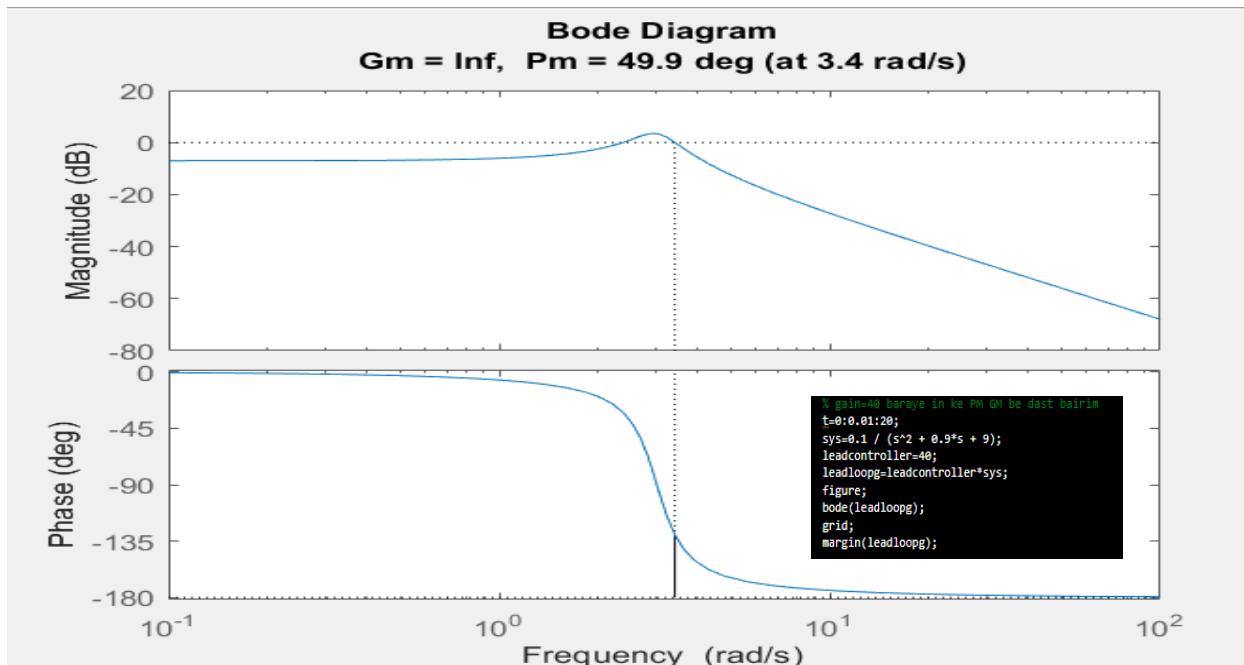
```
System with Controller - Step Response Info:
    RiseTime: 0.0852
    TransientTime: 0.6196
    SettlingTime: 0.6196
    SettlingMin: 0.8100
    SettlingMax: 1.0229
    Overshoot: 13.8483
    Undershoot: 0
    Peak: 1.0229
    PeakTime: 0.2850
```

شکل ۴-۶

همانطور که مشاهده میشود، تمامی خواسته های کنترلی به خوبی برآورده شده است. زمان نشست سیستم به طور قابل توجهی کمتر از ۱۰ ثانیه است، که انتظار می رفت، چرا که استفاده از دو کنترلر **Lead-Min** زیادی سرعت سیستم را افزایش داده است. این موضوع را میتوانید در پاسخ زمانی شکل ۴-۶ و ۴-۷ نیز مشاهده کنید که سیستم دارای کنترلر با شیب خیلی زیادی به شدت ۱ میرود و به عبارتی سرعت این سیستم بهبود یافته است. همچنین، مقدار فرجهش در نبود کنترل حدود  $60^\circ$  است که به  $13^\circ$  کاهش یافته است، این فرجهش در محدوده ای بین  $10\%$  تا  $15\%$  قرار دارد و این نشان دهنده این است که خواسته های کنترلی به خوبی محقق شده اند. توجه نمایید که به دلیل کاهش تایپ سیستم و حذف قطب در مبدأ خطای ماندگار به ورودی پله هرگز صفر نمی شود اما میتوان آن را با افزایش بهره کاهش داد. افزایش بهره در فرکانس های پایین با هزینه هایی همچون نیاز به **control effort** بالا و هزینه ساخت بالا به دلیل تعدد کنترلر ها وجود محدودیت های فیزیکی ممکن است به اندازی کافی بزرگ نشود مصالحه بین خواسته های کنترلی اجتناب ناپذیر است.

حال به بررسی حل دستی می پردازم:

ابتدا نیاز داریم تا به ازای یک gain دلخواه مانند ۴۰ حاشیه فاز و بهره را به دست آوریم برای این کار از یک کد متلب استفاده کرد و نمودار بود آن را رسم می‌کنیم. با استفاده از دستور margin می‌توانیم مقادیر خواسته شده را به دست آوریم. باقی دستورات استفاده شده همه تکراری بوده به همین جهت برای جلوگیری از تکرار آنها را توضیح نمی‌دهیم.



شکل ۷-۶

برآورده شده کنترل فرآیند زیان نسبت استفاده کنیم

$$G(s) = \frac{0.1 \times 40}{s^2 + 0.9s + 9}$$

با این نمودار بود در متاب در راهنمای لذت ۳.۴  $\frac{\text{rad}}{\text{s}}$  در ۴۹.۹ درجه حاصل شد

$$\theta = \frac{\pi \times 40}{0.12} \rightarrow \theta = 55.98^\circ$$

$$\varphi_m = 55.98^\circ - 45^\circ \rightarrow \varphi_m = 10.1^\circ$$

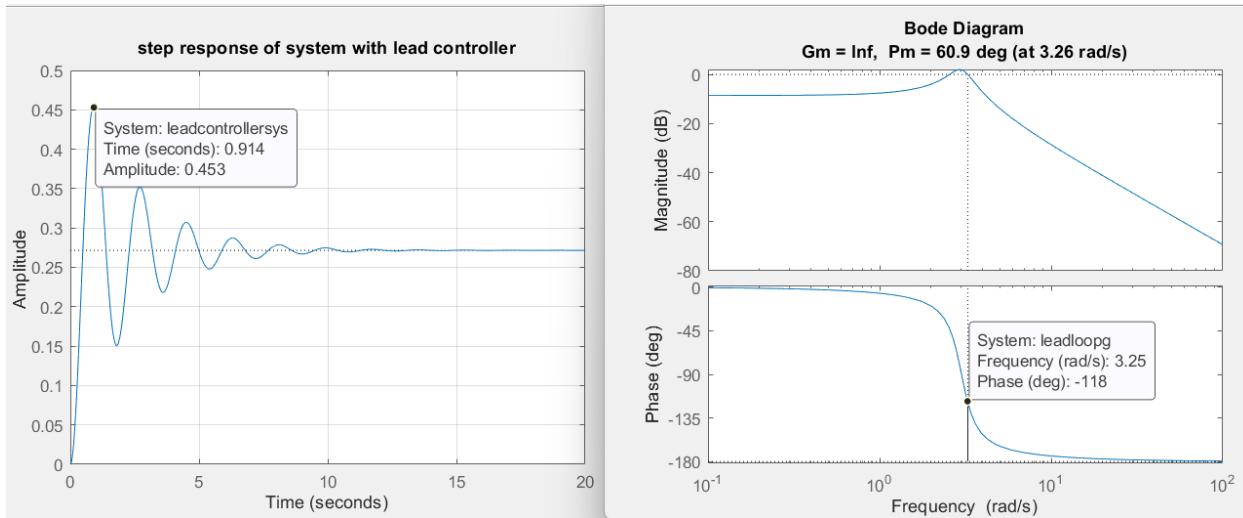
$$\varphi_m = 55.98^\circ - 45^\circ = 10.1^\circ$$

$$\alpha = \frac{1 + \sin \varphi_m}{1 - \sin \varphi_m} = 1.42, T = \frac{1}{\sqrt{\alpha} \omega_c} = 0.246$$

$$C_{پیچیده} = 33.56 \times \frac{1 + 0.349s}{1 + 0.246s}$$

شکل ۸-۶

توجه نمایید که به فاز ۵ درجه در جهت اطمینان اضافه شده است.  
در این مرحله کنترل گر lead اضافه می‌کنیم تا تأثیر آن در نمودار بینیم:



شکل ۹-۶

همان طور که در پاسخ زمانی مشهود است این سیستم دارای فرآجेहش زیاد همچنین خطای ماندگار زیادی است. برای رفع فرآجेहش از یک کنترلگر lag استفاده می‌کنیم. کنترلگر نهایی حاصل ضرب تمامی کنترلگر های به دست آمده تا این مرحله خواهد بود.

```
%نمودار یا نمودار کنترل lag برای مرحله نهایی
t=0:0.01:20;
lagcontroller=1.6*(1.94*s+1)/(3.1*s+1);
leadlagloopg=lagcontroller*leadloopg;
figure;
bode(leadlagloopg);
title('bode diagram of loop gain with lead and lag controller')
grid;
margin(leadlagloopg)
leadlagcontrollersys= leadlagloopg/(1+leadlagloopg);
figure;
step(leadlagcontrollersys , t);
title('step response of system with lead and lag controller')
grid on;
stepinfo(leadlagcontrollersys)
```

*نمودار کنترل lag برای مرحله نهایی*

$$C_{lag} = 1.6 \times \left( \frac{1.94 s + 1}{3.1 s + 1} \right)$$

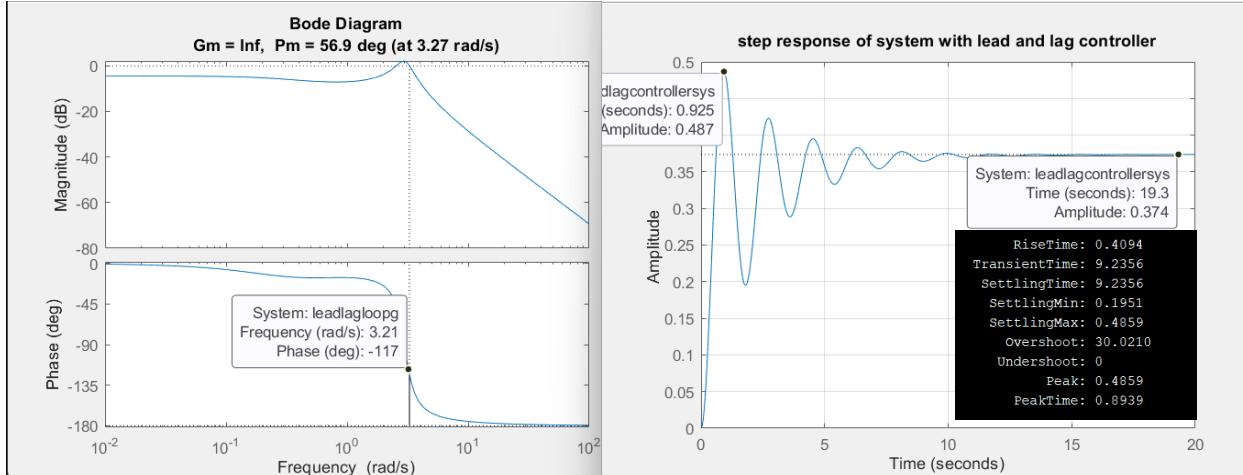
$$\alpha = \frac{1}{k_C} = 0.625$$

$$T = \frac{1}{3.85} \sqrt{\left(\frac{0.6}{0.05}\right)^2 - 1} = 3.1$$

$$C = C_1 \times C_2$$

شکل ۱۰-۶

نمودار بودی حلقه بسته نهایی با حاشیه فاز مناسب و پاسخ پله سیستم نهایی با فرآجेहش بهبود یافته و زمان نشست مطلوب در ادامه قرار می‌گیرد.



شکل ۱۱-۶

## روش شناسی سوال ۷)

در این بخش باید کنترل کننده هایی طراحی کنید که خواسته های کنترلی را ارضاء کنند. برای سیستم تخمین زده شده کنترل کننده ای طراحی کنید که خط ماندگار در تعقیب شیب ورودی را به زیر ۲ درصد برساند. با استفاده ازتابع تبدیل حسابی کنترل کننده ای طراحی کنید که فروژه شیب سیستم به کمتر از ۶ درصد و زمان نشست را به زیر ۶ ثانیه برساند. خروجی نهایی سیستم و رفتار سیگنال کنترلی را به ازای ورودی شیب و پله نمایش دهید.

در سوالات قبل تابع تبدیل این سیستم تخمین زده شد، دیدیم که دارای یک قطب در مبدأ است یا به عبارتی یک سیستم تایپ ۱ است. در نتیجه خطای این سیستم به دلیل وجود یک انتگرال گیر ذاتی به ورودی پله ۰ است و به ورودی شیب مقدار خطای محدود دارد و مقدار عددی آن قابل محاسبه است. در ادامه با حل دستی این مقدار محاسبه شده است.

$$\lim_{s \rightarrow 0} sG(s) = \lim_{s \rightarrow 0} \frac{s(0.1s - 0.2)}{s^2 + 0.9s + 9} = -\frac{2}{90}$$

$$K = \frac{45}{0.02} = 2250$$

خطای حالت صاف‌کار حقیقتی و خودکار بهتر منتهی‌نماید. اول باید یعنی منتهی‌بودن تا مزدادر

باشد باید. باید آنچه خطای بزرگتر ۲٪ برسانیم یعنی یعنی روزی بیسیم. این لیست زیر را عنوانی می‌بریم

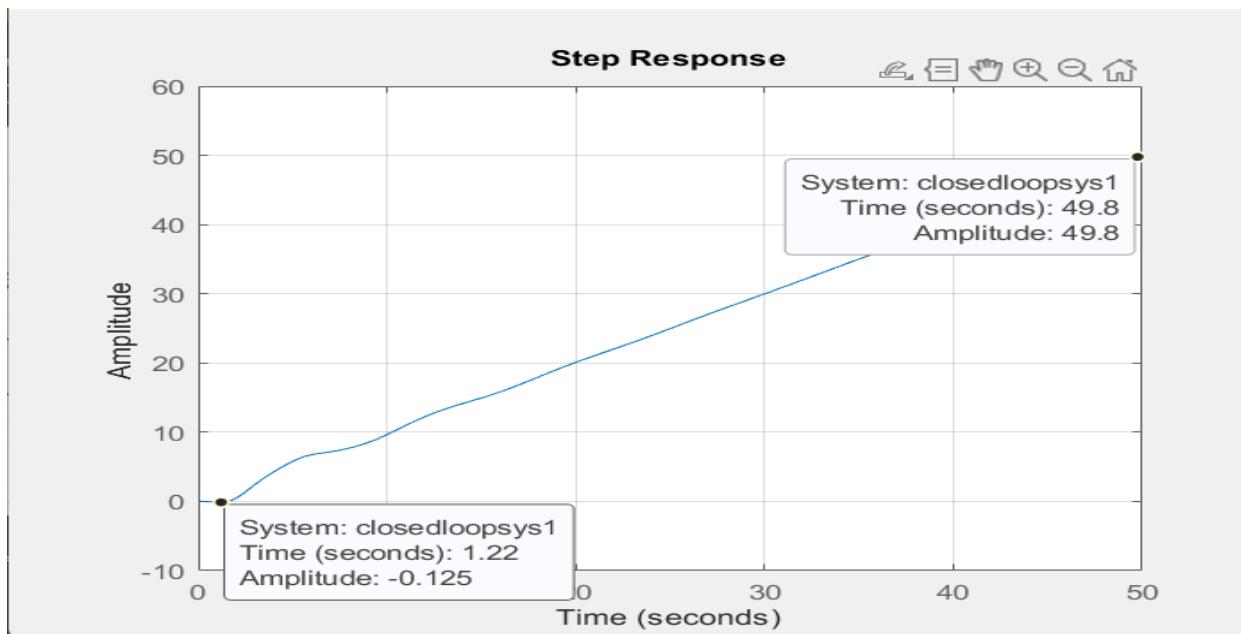
تحمیل کرد چنان‌که ممکن‌لت نهاده ارائه بسیار باید دو عدد lag مکردم. این کار در سیمولین

دد ۲۵ اینچ در طبقه غردن در زمان ۴۰ تا ۵۰ ثانیه بروی تسیبل در این خطای صاف‌کار نماین

. است.

شکل ۱-۷

برای رسم پاسخ سیستم به ورودی شیب می‌توانیم از ضرب  $s/1$  در سیستم استفاده کنیم و سپس از سیستم خروجی پله بگیریم که با شیب معادل خواهد بود.



شکل ۲-۷

برای از بین بردن خطای ماندگار می‌توانیم گین را افزایش‌ش دهیم مشکل اینجاست که سیستم‌های دارای محدودیت‌های فیزیکی هستند و به عبارتی نمی‌توانند گین‌های خیلی بزرگ را ایجاد کنند. به همین دلیل با استفاده از ابزار sisotool به وسیله PIDtunning به جای گین بالا از کنترلگر صنعتی pi استفاده می‌کنیم که باعث کاهش خطای ماندگار می‌شود. همانطور که در شکل ۲-۷ مشهود است ابتدا سیستم دارای خطأ است اما با گذشت زمان این خطأ به زیر ۲٪ کاهش می‌ابد.

### استفاده از قاعده حاست:

تابع حاست دارای یک متر غیرکنstant ناز است که این متر را متر ۲ می‌کند.

از قاعده تکدیک این متر بسیار کمتر متر را متر ۲ می‌کند. جن این بنت ۲

$$T_d = \frac{\frac{S}{2} + 1}{(S+1)^2}$$

$$T_d(2) = 0 \Rightarrow T = \underline{\underline{2}}$$

$$T_d = \frac{\frac{-S+1}{2}}{(S+1)^3}$$

بنابراین کدام تبدیل متحصل است؟

$$C = \frac{kT}{SXP} = \frac{k(S^3 + 0.9S^2 + 9S)}{S^3 + 3S^2 + 3.5S}$$

کنترلر سنسور:

شکل ۲-۷

با استفاده از ابزار sisotool در متلب تلاش می‌کنیم گینی را پیدا کنیم که در آن خواسته‌های کنترلی سوال به درستی برقرار شود، توجه نمایید که سیستم ما دارای یک صفر غیرکمینه فاز است پس در نتیجه برای پایدار شدن نیاز به گین منفی دارد. خواسته‌های کنترلی ما با گین -۵.۶۸۸- برقرار شدند که در ادامه کد مربوطه و خروجی‌ها خدمتتان ارسال می‌گردد.

```
%in bakhsh baraye nakhshe dovom tarahi ba estefade az controller tabee
%hasasiat
secondcontroller = (-5.688 * (S^3 + 0.9*S^2 + 9*S)) / (S^3 + 3*S^2 + 3.5*S);

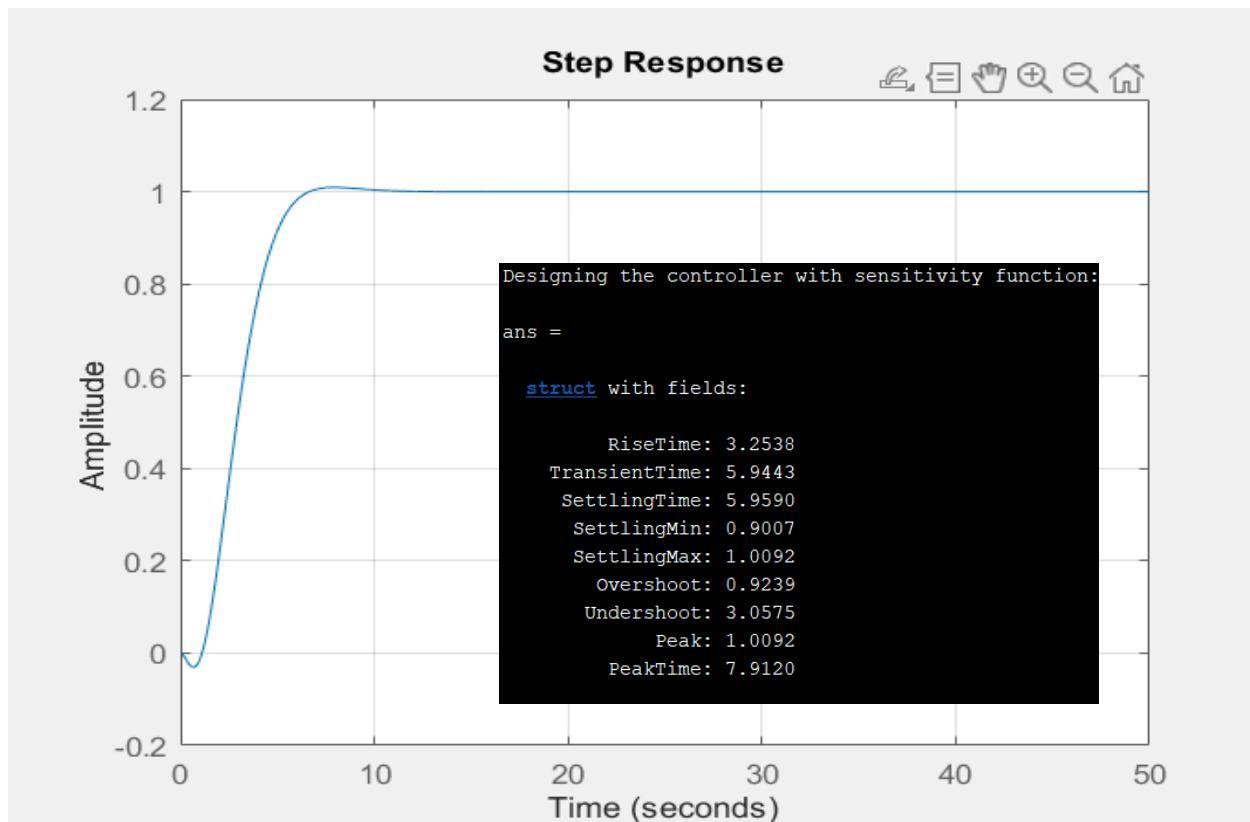
secondloopg = secondcontroller * sys;
closedloopsys2 = secondloopg / (1 + secondloopg);

figure;
step(closedloopsys2, t); % step response baraye system dovom
grid;

fprintf("\nDesigning the controller with sensitivity function:\n");
stepinfo(closedloopsys2) % Info-hayi mesle rise time, settling time ro neshun midahad
```

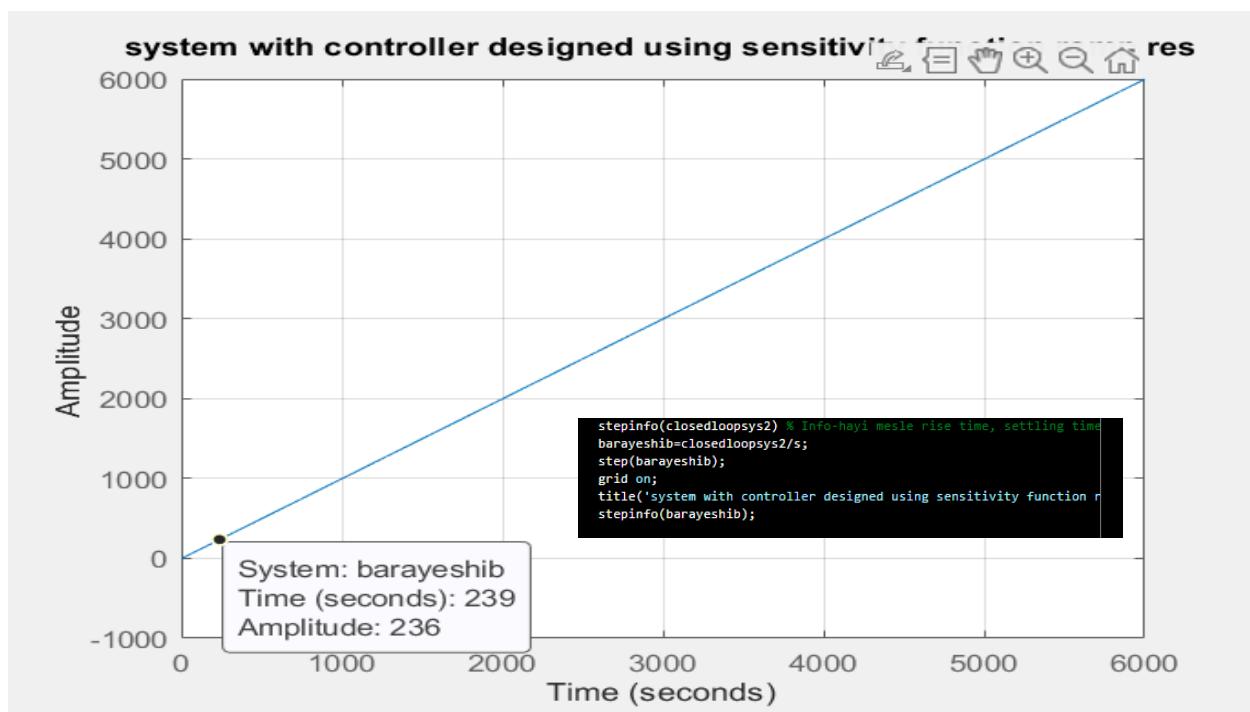
شکل ۴-۷

خروجی این کد در شکل بعدی آورده شده خواهد دید که با این گین اهداف کنترلی ما برقرار می‌شود، سپس به بررسی control effort می‌پردازیم. انتظار می‌رود خطای ماندگار به ورودی پله صفر و به ورودی شیب عددی محدود باشد که سعی بر کمینه کردن آن داریم.



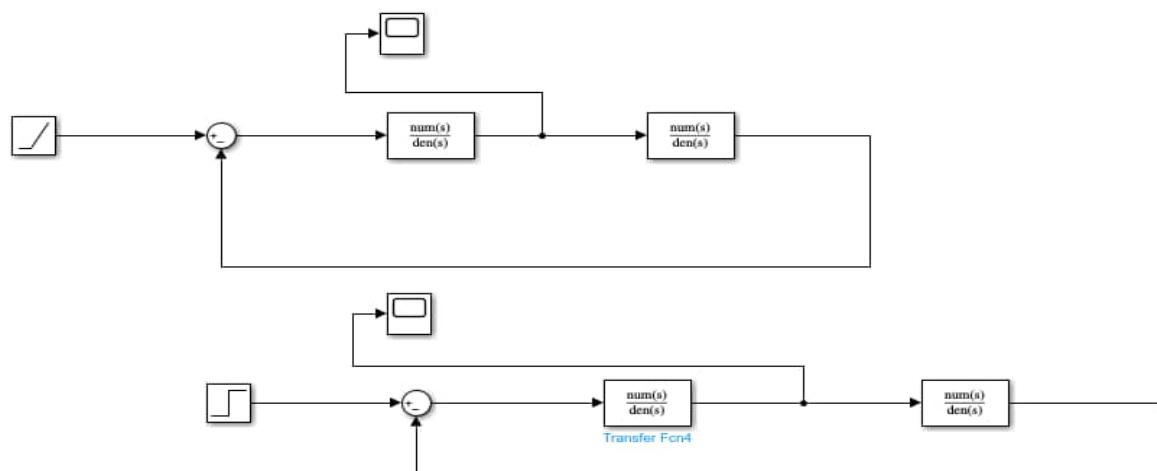
شکل ۵-۷

این سیستم دارای فروجہش کمتر از ۶٪ و زمان نشست زیر ۶ ثانیه است. همان طور که انتظار میرفت خطابه ورودی پله ۰ است و همچنین وروردی شب را به خوبی تعقیب می کند، توجه نمایید به دلیل درصد کم خطاب مطلب آن را نمایش نداد اما مقدار خیلی کمی خطاب دارد.

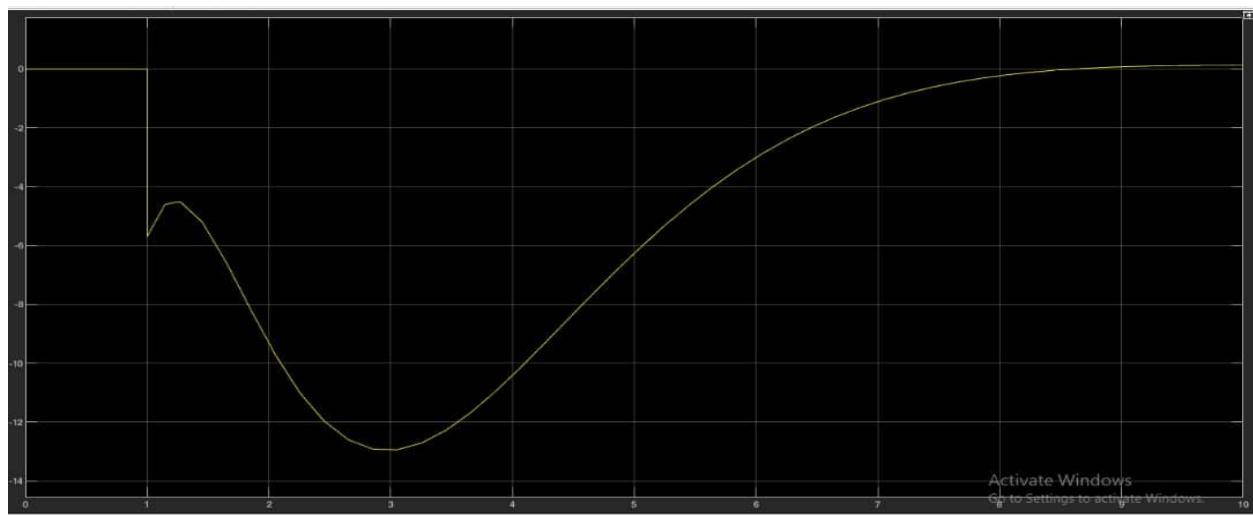


شکل ۶-۷

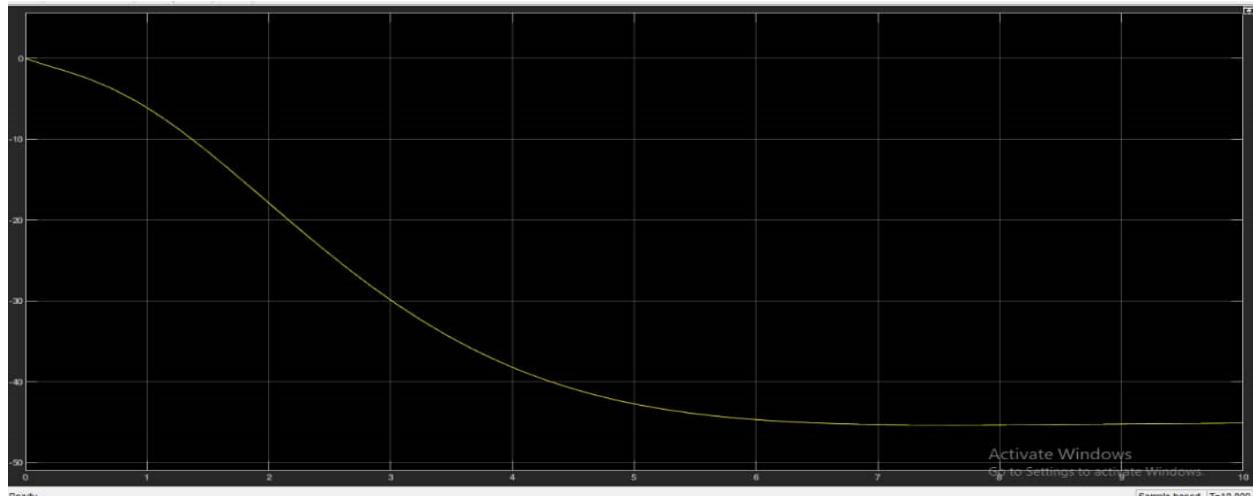
در ادامه با استفاده از Simulink نرم افزار مطلب می توانیم برای هر دو ورودی پله و شیب تلاش کنترلی را بررسی کنیم . در ادامه بلوک دیاگرام طراحی شده به این منظور و نمودار مربوط به control effort خدمتتان ارسال می کنیم.



شکل ۷-۷



شکل ۸-۷



شکل ۹-۷

نمودار های موجود در شکل ۸-۷ و ۹-۷ مربوط به خروجی بلوك دیاگرامی است که در شکل ۷-۷ مشاهده می‌کنید. control effort که در شکل ۸-۷ می‌بینید برای ورودی پله است همان طور که انتظار می‌رفت تلاش کنترلی مورد نیاز برای صفر کردن خطای پله به صفر میل می‌کند چرا که سیستم تایپ ۱ بود و در زمان بینهایت یا همان حالت ماندگار به ورودی پله خطای صفر خواهد داشت و نیازی به گین برای از بین بردن این خطأ و به عبارتی تلاش کنترلی ای نداریم اما در در نمودار دوم می‌بینید که برای ورودی شبیه همانطور که انتظار می‌رفت نیاز به تلاش کنترلی داریم چرا که سیستم های تایپ ۱ به ورودی شبیه خطای ماندگار محدودی می‌دهند که برای کمینه کردن آن گین سیستم را افزایش میدهیم این عمل نیازمند تلاش کنترلی است. از شکل ۹-۷ می‌توان نتیجه گرفت این تلاش کنترلی به مقداری ثابت میل می‌کند و خطای سیستم به ورودی شبیه در حالت ماندگار مقداری ثابت خواهد شد. این مقدار طبق تحلیل های قبلی بسیار کوچک است.

در نهایت از وقت گرانقدر شما که در جهت مطالعه این گزارش صرف کرده اید کمال تشکر را داریم.