

«به نام خداوند جان و خرد»



دانشگاه صنعتی خواجه نصیرالدین طوسی

نام و نام خانوادگی: مبینا یوسفی مقدم

شماره دانشجویی: ۴۰۱۲۴۰۹۳

موضوع: گزارش پروژه پایانی

درس سیستم‌های کنترل خطی

نام استاد: جناب آقای دکتر تقی‌راد

بهمن ماه ۱۴۰۳

چکیده

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

روش شناسی 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 از داده‌های موجود، دیاگرام بودی سیستم را رسم کنید.

در این سوال، در تلاشیم با استفاده از نرم افزار 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 راحت تر 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 متغیر های یاد شده ذخیره می‌کنیم تا به توانیم نمودار های خواسته شده را رسم کنیم.

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

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

شکل ۲-۱

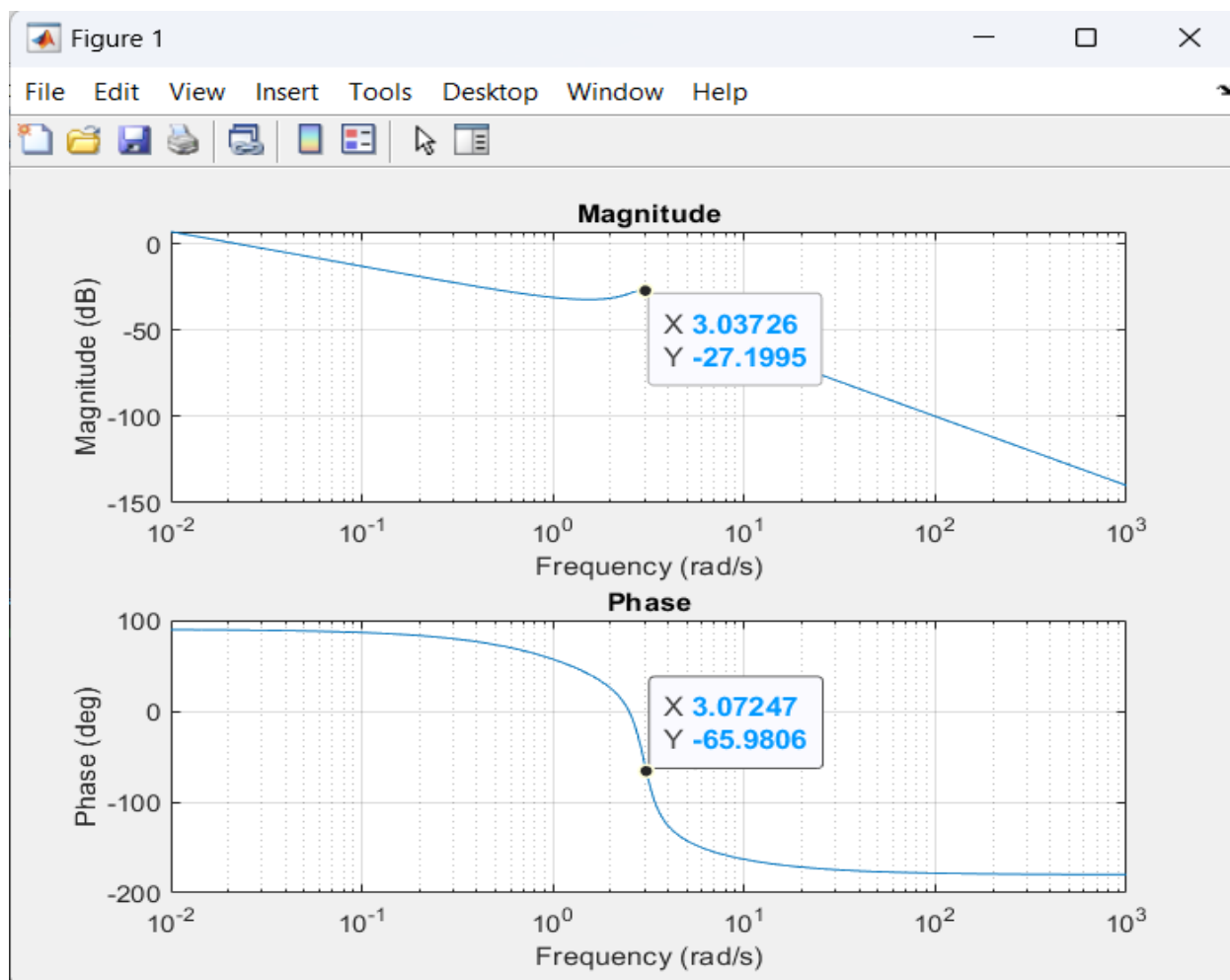
هدف این بخش کد رسم نمودار دامنه و فاز است. با استفاده از subplot میتوان در یک صفحه نمودار ها را نمایش داد که تحلیل و مقایسه را سهولت می‌بخشد. همچنین با رسم نمودار دامنه و فاز در اسکیل لگاریتمی میتوان محتوای بیشتری از داده را در یک صفحه نمایش داد. این به معنی شناخت بهتر سیستم به دلیل اطلاعات بیشتر از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

محور افقی لگاریتمی باشد. توجه به فرماید 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 می‌توانیم 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 nabashe khabar mide ke
% file moshkel dare
```

شکل ۳-۱

همان طور که قبلا 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 گرفته‌ایم.



شکل ۴-۱

در شکل ۴-۱ خروجی مربوط به سوال ۱ مشاهده می‌شود که شامل دامنه و فاز است. هر دو نمودار همان طور که پیشتر توضیح داده شد بر مبنای لگاریتمی رسم شده اند. دامنه برحسب db و فاز برحسب درجه است. همانطور که می‌بینید در فرکانس ۳ رادیان بر ثانیه دامنه مقداری افزایش داشته این به دلیل حضور صفر غیرکمینه فاز است. هرچه زتا یا همان damping ratio کاهش یابد مقدار این جهش بیشتر می‌شود و در صورت تحریک سیستمی با زتا کوچک در فرکانسی که این جهش رخ می‌دهد، 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 می‌آورد.

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 شیب منفی ۲۰ دسی‌بل آغاز می‌شود. در نمودار فاز نیز مشاهده می‌شود که فاز از مقدار مثبت ۹۰ درجه شروع می‌کند، در حالی که باید از منفی ۹۰ درجه آغاز می‌شد. این تغییر فاز به اندازه ۱۸۰ درجه نشان‌دهنده تأثیر یک تغییر فاز ۱۸۰ درجه‌ای در سیستم است.

در ادامه، اگر به نمودار اندازه توجه کنیم، در فرکانس ۲ رادیان بر ثانیه مشاهده می‌کنیم که نمودار تقریباً به ثبات می‌رسد و افزایش ۲۰ دسی‌بل در سیستم نمایان می‌شود. این تغییرات نشان‌دهنده وجود یک صفر در سیستم در این فرکانس است. در فرکانس ۳ رادیان بر ثانیه، افزایش ناگهانی در اندازه دیده می‌شود و همزمان تغییرات قابل توجهی در فاز اتفاق می‌افتد. در این نقطه، فاز به مقدار ۱۸۰ درجه کاهش می‌یابد که به وضوح نشان‌دهنده حضور یک قطب مرتبه دوم در این فرکانس است. از سوی دیگر، کاهش فاز ۹۰ درجه‌ای نیز به علت وجود یک صفر غیر کمینه فاز در سیستم است.

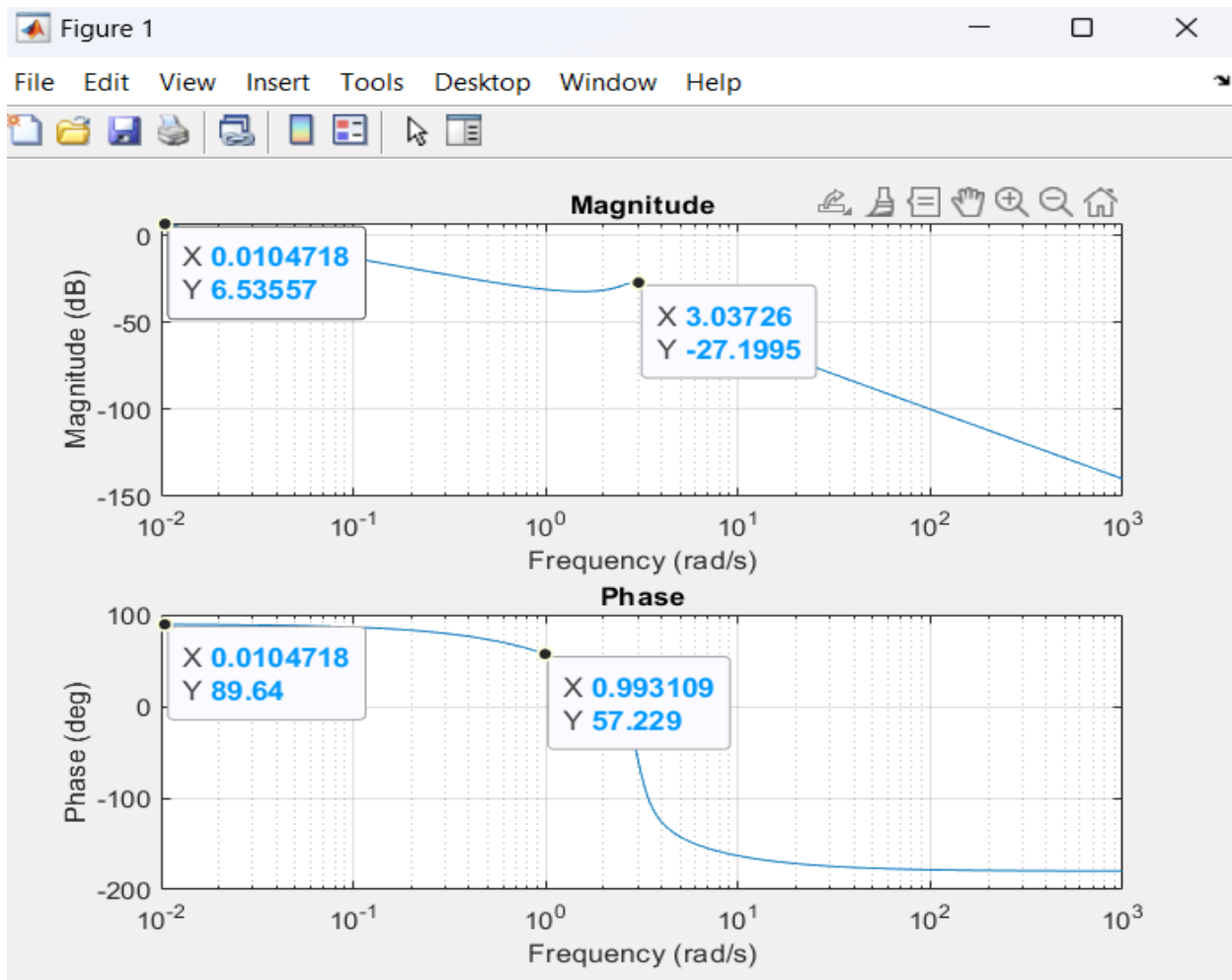
در ناحیه فرکانس‌های بالا، مشاهده می‌کنیم که نمودار اندازه با شیب منفی ۴۰ دسی‌بل ادامه می‌یابد. این وضعیت نشان‌دهنده آن است که تعداد قطب‌ها ۲ تا بیشتر از صفر هاست. بنابراین، وجود یک صفر و سه قطب در سیستم منطقی به نظر می‌رسد. فرکانس‌های مربوط به صفرها و قطب‌ها نیز از نمودار به وضوح قابل شناسایی است.

در خصوص نوع سیستم، از آنجا که نمودار اندازه با شیب منفی ۲۰ دسی‌بل شروع می‌شود، می‌توان نتیجه گرفت که سیستم از نوع تیپ یک است و بیانگر وجود یک قطب در مبدا 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 ۳ خواهد بود.

تأخیر سیستم معمولاً از طریق تغییرات فاز نسبت به تغییرات فرکانس در نمودار نشان داده می‌شود. هرچه این تغییرات فاز بیشتر و شدیدتر باشد، تأخیر سیستم افزایش می‌یابد، و هرچه تغییرات فاز کمتر باشد، تأخیر سیستم کاهش پیدا می‌کند. برای محاسبه تقریبی این تأخیر، ابتدا باید شیب خط وصل‌کننده دو نقطه از نمودار را محاسبه کنیم. برای این منظور، مقدار فاز را در فرکانس ۰/۱۰۴ که تقریباً می‌توان آن را برابر با صفر در نظر گرفت، خوانده و سپس همین کار را برای فرکانس ۰/۹۹ که تقریباً همان ۱ رادیان بر ثانیه است نیز انجام می‌دهیم (شکل ۲-۱). پس از این مرحله، شیب 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 تأخیر سیستم را بر اساس این تغییرات فاز محاسبه خواهیم کرد. نتیجه این محاسبات نشان می‌دهد که تقریباً نیم ثانیه تأخیر در سیستم وجود دارد.

این سیستم غیر کمینه فاز است. همان‌طور که پیش‌تر اشاره شد، در فرکانس ۲ رادیان بر ثانیه، نمودار اندازه به طور تقریبی ثابت می‌شود که نشان‌دهنده افزایشی ۲۰ دسی‌بل به شیب منفی ۲۰ 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 غیر کمینه فاز در سیستم است، صفرهایی که باعث کاهش مقدار فاز در این فرکانس‌ها می‌شوند.



شکل ۱-۲

در ادامه محاسبات توضیحات فوق ارائه خواهند شد:

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

$$\omega = 0 \rightarrow \phi = 89.64$$

$$\omega = 1 \rightarrow \phi = 57.229$$

توجه فرمایید فرکانس های فوق تقریبی هستند.

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

این سیستم تقریباً با نیم ثانیه تاخیر واکنش نشان می‌دهد.

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

روش شناسی سوال (۳)

با توجه به پاسخ فرکانسی داده شده، به سیستم یک تابع تبدیل مناسب برازش 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 برازش کنید.

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

در ابتدا، نمودار بامیله $-20 \frac{db}{decade}$ - کاهش میله پس یک قطب در مبدأ داریم. در فرکانس

2 rad/sec فاز سیستم کاهش بیستی داشته. ریب باند $20 \frac{db}{decade}$ رو با افزایش است. بنابراین یک

صفر میله فاز در این فرکانس داریم. همچنین در 3 rad/sec تغییر ناگهانی زیاد شدن نود انداز را داریم و نمودار

فاز نیز در این فرکانس به قدر ناگهانی به مقدار زیادی کاهش پیدا کرده است. پس یک قطب در 2 rad/sec داریم.

$$\omega_1 = 2 \text{ rad/s} \rightarrow z_1 = 0.5 \quad \text{و} \quad \omega_2 = 3 \text{ rad/s} \rightarrow z_2 = 1.5$$

مقدار ζ را نیز با فرض دگرزما شدن روی نمودار بدست می‌آوریم. با توجه به اینکه فاز بامیله بسیار زیادی کاهش

پیدا کرده، مشخص است که ζ صابیت نزدیک صفر است. $\zeta = 0.15$

$$L(s) = \frac{1 - 0.5s}{s(1/9s^2 + 2/3 \times 0.15s + 1)} = \frac{1 - 0.5s}{s(1/9s^2 + 0.1s + 1)}$$

حال باید گفت به آن افزوده شود تا تابع تبدیل خنوع به بلاک شده شود، مطابق بودی سیستم شود.

با ضرب مقادیر مختلف می‌بینیم که باید 0.1 در آن ضرب شود:

$$L(s) = 0.1 \times \frac{1 - 0.5s}{s(1/9s^2 + 0.1s + 1)}$$

شکل ۱-۳

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

در سوال ۳-۱ 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 System Identification را در Command Window باز کرده و از آنجا با تعیین تعداد صفرها و قطب‌ها، همچنین مشخص کردن پیوسته یا گسسته بودن سیستم، تابع 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 به‌طور 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** به‌صورت 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 **tfest** یا **arx** 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 تخمین زده شده را ارزیابی کنیم.

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

```

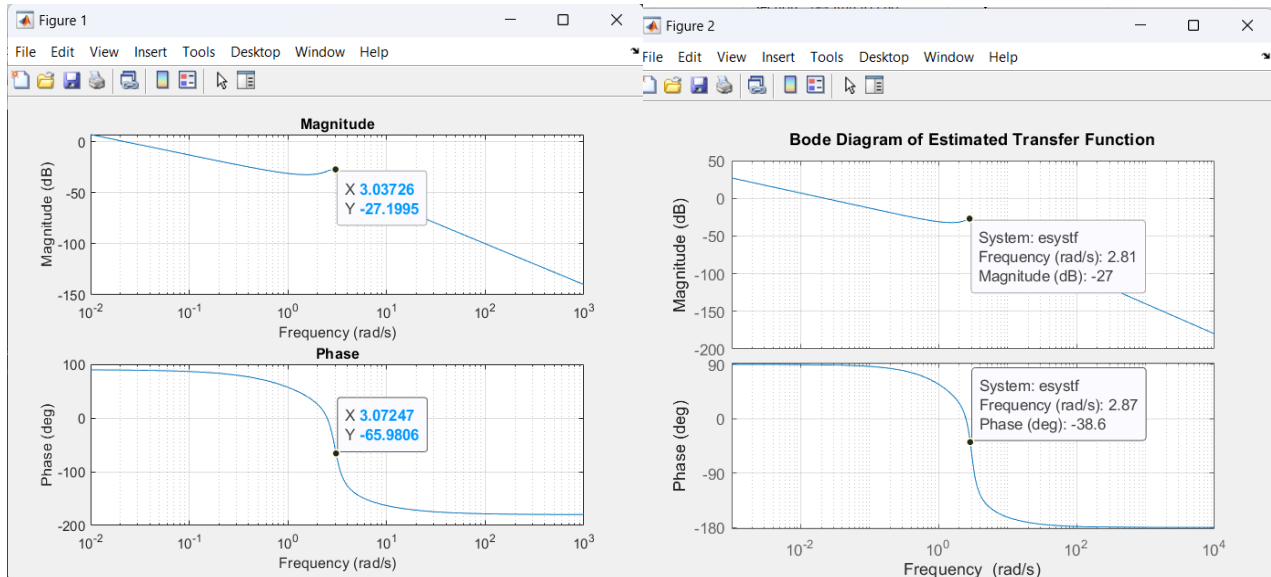
1 data = load('Data.mat');
2 magnitude = data.Data.magnitude;
3 phase = data.Data.phase;
4 frequency = data.Data.omega;
5
6 ghotbi = magnitude .* exp(1j * deg2rad(phase)); %form ghotbi
7
8
9
10 freqresdata = idfrd(ghotbi, frequency, 0); % Ts = 0 tebgh chat gpt chon system continuous bood ts=0 va agar 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 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 completly empty');
25 end
26 % dar soorati ke file data.mat dakhelesh on etelaat nabashe khabar mide ke
27 % file moshkel dare

```

شکل ۳-۲

به جهت صرفه‌جویی در زمان بخش‌های تکراری کد موجود در شکل ۳-۲ بررسی نخواهد شد. توضیحات را مشترکا در روش شناسی سوال ۱ خدمتان داده‌ایم. برای این که بتوانیم پاسخ فرکانسی سیستم را با توجه به داده‌های مسئله به دست بیاوریم ابتدا آنها 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 آوردیم می‌توانیم تابع تبدیل سیستم را تخمین بزنیم. با مقایسه دامنه

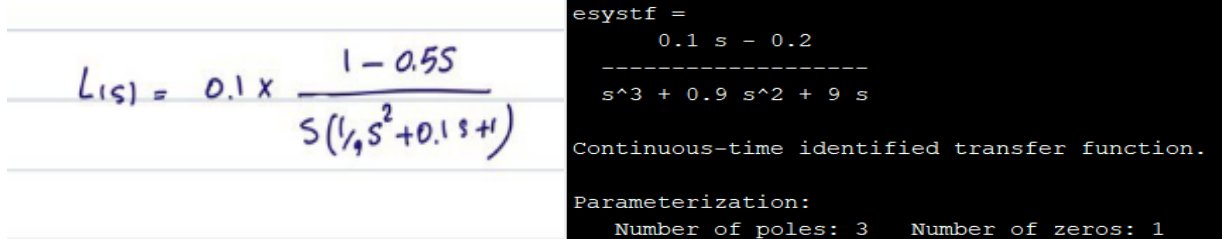
و فاز سیستم تخمین زده شده با شکل ۴-۱ میبینیم که شباهت زیادی دارد و این به طر کلی تابع تبدیل به دست آمده در شکل ۱-۳ و ۲-۳ با تقریب خوبی یکسان اند.



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.



شکل ۵-۳

شکل ۴-۳

روش شناسی سوال (۴)

با استفاده از معیار پایداری راث-هرویتز مشخص نمایید که به ازای 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 را در شکل ۱-۴ مشخص کرده‌ایم. از اشتراک حالات قابل قبول ناحیه ای برای k به دست می‌آید که تمامی شروط را ارضا می‌کند.

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

استفاده از معیار پایدار را به صورت:

$$C = K$$

$$C X S / S - 1 = K \times \frac{0.15 - 0.2}{s^3 + 0.9s^2 + 9s} \rightarrow T(s) = \frac{K \left(\frac{0.15 - 0.2}{s^3 + 0.9s^2 + 9s} \right)}{1 + K \left(\frac{0.15 - 0.2}{s^3 + 0.9s^2 + 9s} \right)}$$

پس از ساده سازی:

$$T(s) = \frac{0.15 - 0.2}{s^3 + 0.9s^2 + (9 + 0.1K)s - 0.2K}$$

حال از معیار پایدار استفاده کنیم:

| | | | |
|-------|----------------------|------------|--|
| s^3 | 1 | $9 + 0.1K$ | |
| s^2 | 0.9 | $-0.2K$ | |
| s^1 | $8.1 + 0.09K + 0.2K$ | 0 | |
| s^0 | $-1.62K - 0.058K^2$ | 0 | |

شرط اول: $8.1 + 0.09K + 0.2K > 0 \rightarrow K > -27.931$

شرط دوم: $-1.62K - 0.058K^2 > 0 \rightarrow$ $\begin{cases} \text{مثبت} & K > 0 \\ \text{مثبت} & K < 0 \end{cases}$

پس برای ناحیه پایدار می باشد:

$$-27.931 < K < 0$$

شکل ۱-۴

روش شناسی سوال ۵

مکان هندسی سیستم بدست آمده را رسم کنید. با استفاده از این نمودار تحلیل کنید که آیا می توان با استفاده از کنترل کننده تناسبی سیستم را به پایداری رساند؟ توضیح دهید
با استفاده از کدام یک از کنترل کننده های صنعتی PI و PD می توانیم سیستم را به پایداری برسانیم؟ توضیح دهید.

در این سوال ابتدا به رسم مکان هندسی ریشه ها به ازای تمامی مقادیر k می پردازیم. برای ای کار از تابع تبدیل تخمین زده شده در سوال ۳ استفاده خواهیم کرد. در نهایت تاثیر کنترلر بر سیستم حلقه بسته جدید را با استفاده از نمودار بودی بررسی خواهیم کرد.
از آنجایی که دسترسی به اطلاعات فایل data.mat قبلا در روش شناسی سوال ۱ و همچنین تخمین سیستم از طریق system identification در روش شناسی سوال ۳ توضیح داده شده است، از تکرار این مطالب خودداری خواهیم کرد.
در این کد برای رسم مکان هندسی به جای تابع rlocus از rlocusplot استفاده کرده ایم زیرا در ورژن matlab من grid تنها روی این تابع جواب می داد. مکان هندسی ریشه ها به ازای k مثبت با خط ممند و مکان هندسی ریشه ها 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 شده قرینه آن بدهیم.
توضیحات فوق مربوط به کد موجود در شکل ۵-۱ می باشد.

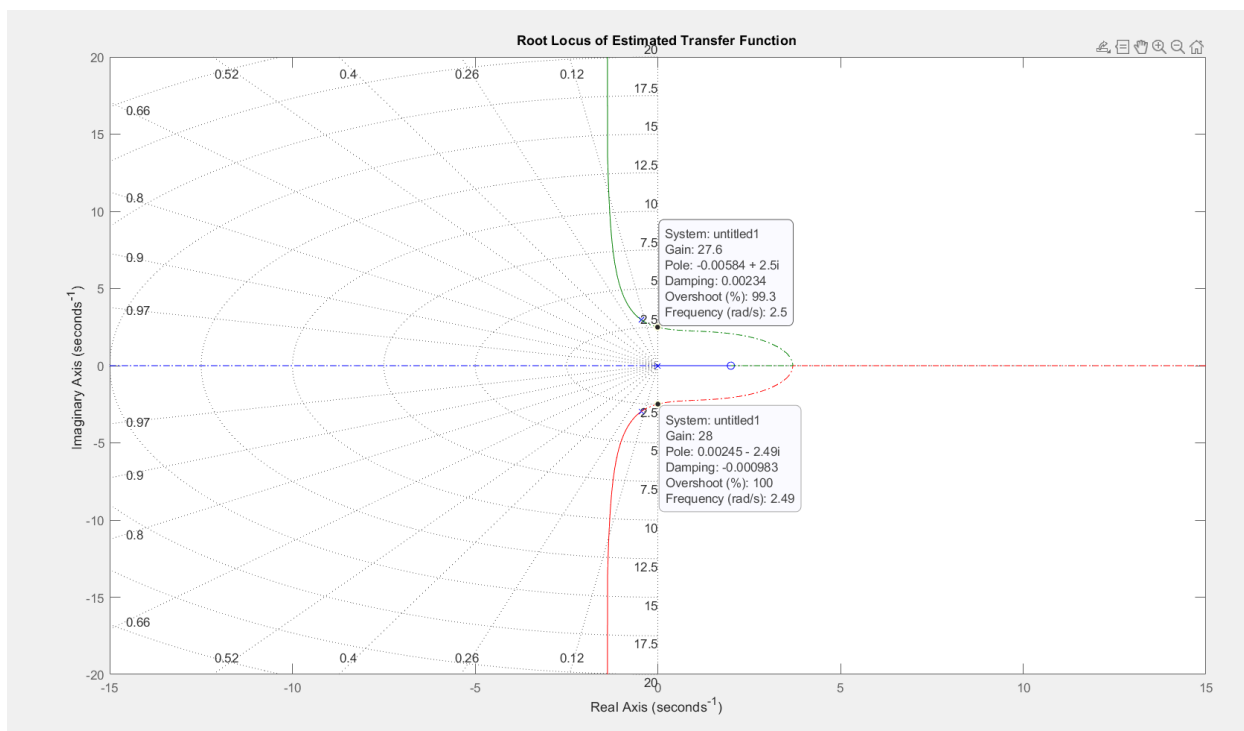
```

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 ghotbi ha yek transfer function misazad
14 disp('Estimated Transfer Function:');
15 disp(esystf);
16
17
18
19 figure;
20 rlocusplot(esystf); %in nemoodar makan hendesi 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 ۵-۱

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 $-27/93$ یا به عبارتی سیستم به ازای $k = -27/93$ در مرز ناپایداری قرار دارد. این موضوع را مکان هندسی ریشه های سیستم تخمین زده شده نیز تایید می‌کند. در ادامه خروجی کد موجود در شکل ۵-۱ را خدمتان ارائه می‌کنیم.



شکل ۵-۲

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. مقتضیات توجهتان را به داده های نمایش داده شده جلب کنیم، می‌بینید که با افزایش اندازه k میزان فرجهش به ۱۰۰٪ یا همان نوسانی کامل افزایش می‌یابد و با ادامه این روند خروجی و اگر خواهد شد و سیستم ناپایدار می‌شود. در این فرایند 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 ناپذیری ایجاد کند البته که این به معنای مطلقاً نامطلوب بودن این شرایط نیست، در بسیاری 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 بسنجیم. 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.

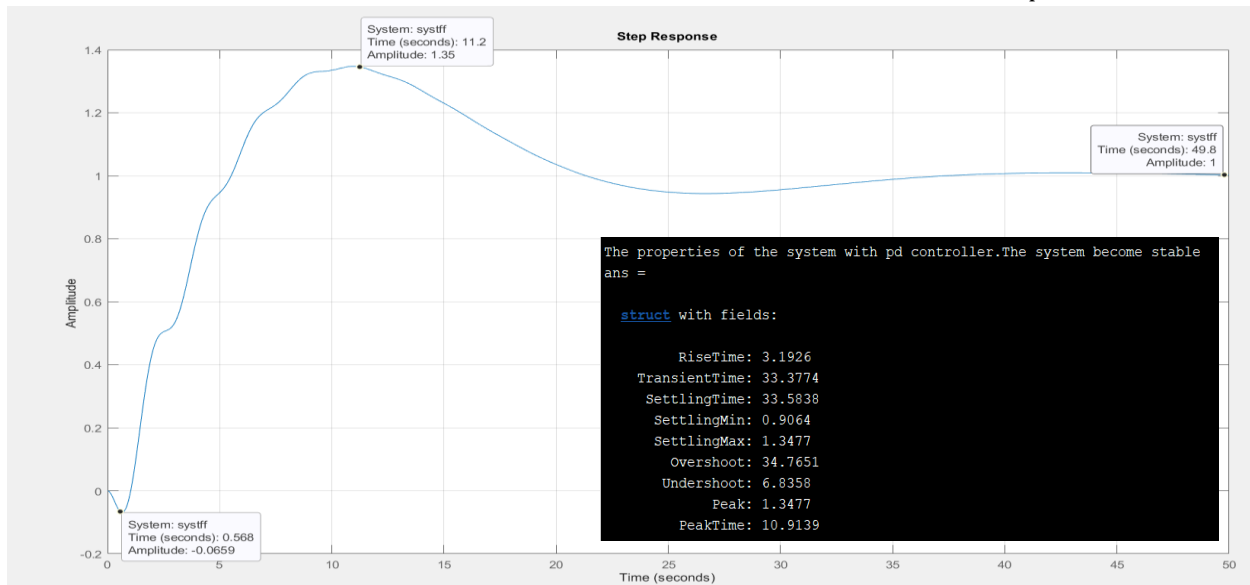
```

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 ghotbi ha yek transfer function misazad
14 disp('Estimated Transfer Function:');
15 disp(esystf);
16
17
18
19 figure;
20 rlocusplot(esystf); %in nemoodar makan hendesi rishe transfer function tarahi shodas
21 %grid in dorost kon
22 %in dastor rlocusplot ejaze grid mide rlocus gridesh faal
23 %nemishod
24
25 grid on;
26 title('Root Locus of Estimated Transfer Function');
27 hold on;
28 rlocusplot(-esystf, '-.'); %root locus be ezaye K<0 ham rasm kone
29 grid on;
30
31 s=tf('s');
32 time=0:0.0001:50;
33 controller=-10*(s+0.2)/s;
34 loopgain=controller*esystf;
35 systff=loopgain/(1+loopgain);
36 figure;
37 step(systff,time);
38 grid;
39 fprintf("The properties of the system with pd controller.The system become stable");
40 stepinfo(systff)
41
42 if ~isfield(data, 'Data') || ~isfield(data.Data, 'magnitude') || ~isfield(data.Data, 'phase') || ~isfield(data.Data, 'omega')
43     error('Required data are missing in the data file or data file is completly empty');
44 end
45 % dar soorati ke file data.mat dakhelesh on etelaat nabashe khabar mide ke
46 % file moshkel dare

```

شکل ۳-۵

در شکل فوق توجه شما را به کد های موجود در خط ۳۰ الی ۳۹ جلب می‌کنیم. ابتدا تابع تبدیل کنترلر 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 `rise time`، `overshoot` 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 ۳۰٪ فرافروش و حدود ۶٪ فروفروش با توجه به همگرایی آن میتوان گفت که سیستم با کنترلر 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 مبدا نداشته باشد. سپس برای این سیستم کنترل کننده ای طراحی کنید که فرافروشی بین ۱۰ تا ۱۵ درصد داشته باشد و زمان نشست هم کمتر از ۱۰ ثانیه باشد. پس از طراحی این کنترل کننده، خروجی سیستم را 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 فرافروش آن را کاهش دهیم. برای رسیدن به این هدف، از دو کنترلر $Lead$ استفاده کردیم. دلیل این انتخاب این است که کنترلرهای $Lead$ 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 توانسته فرافروش 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 فرافروش 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 پایین‌تر از ۰ دسی‌بل قرار داشت لذا نیاز مند به کنترلی 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 کنترلی، نمی‌توانستیم گین را بیش از حد بالا ببریم. و به عبارتی $control\ effort$ خیلی بالا می‌رفت و پیاده سازی آن به دلیل محدودیت های 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.

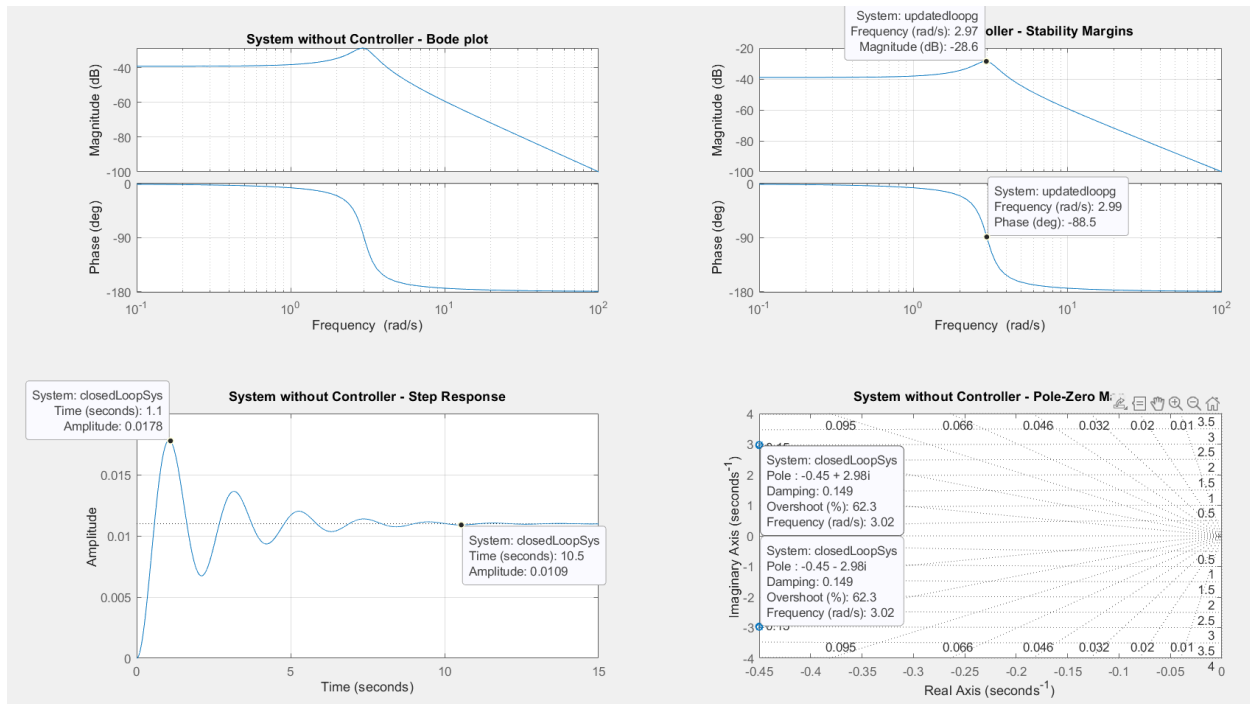
```

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 mizanim 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 تاثیر اضافه شدن کنترلر به سیستم نوشته شده است. به وسیله این کد می‌توان دامنه و فاز سیستم حلقه باز بدون کنترلر و پاسخ پله و محل قرار گیری صفر و قطب های سیستم حلقه بسته را مشاهده کرد.



شکل ۲-۶

همانطور 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 تر آن 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 ها رخ می‌دهد. در ادامه می‌توانید اطلاعات اطلاعات مربوط به پاسخ پله قبل از اضافه کردن کنترلر 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
```

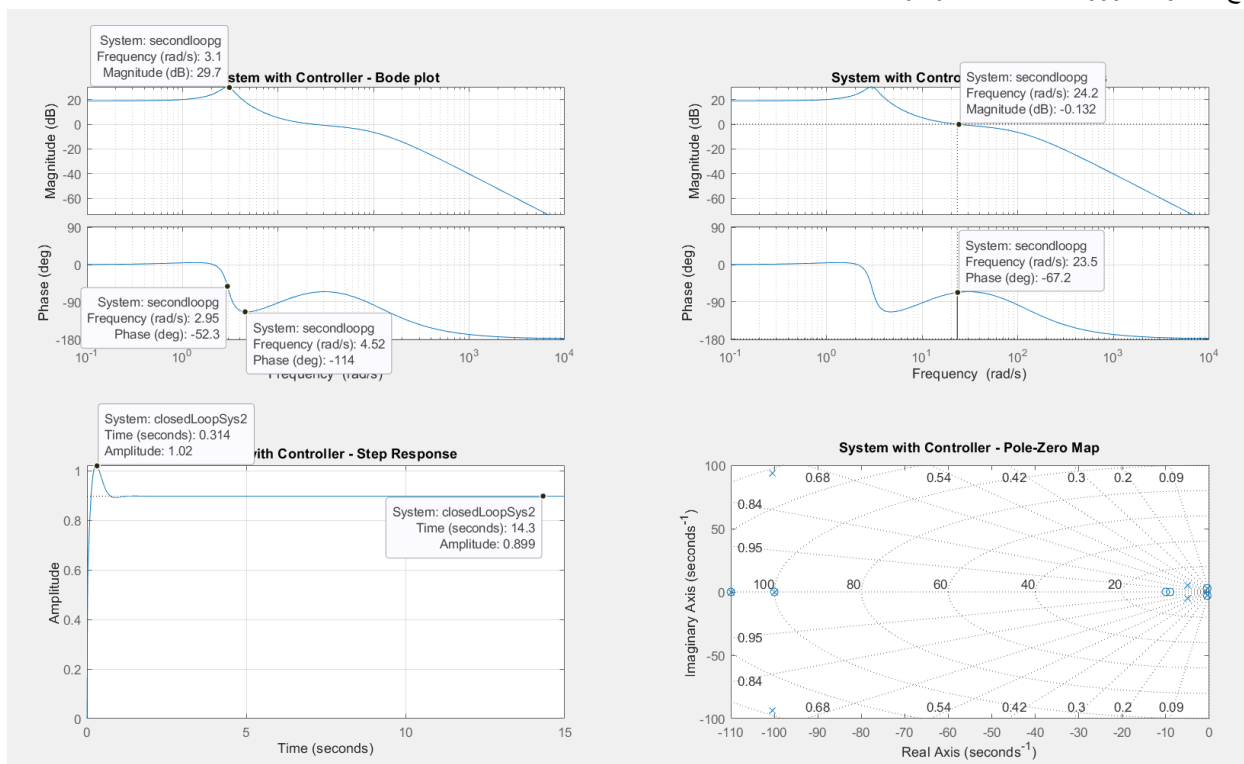
شکل ۳-۶

حال کنترلر را در مسیر پیشرو قرار می‌دهیم و تابع تبدیل سیستم حلقه بسته را به دست می‌آوریم. نتایج حاصل را با شکل ۲-۶ مقایسه خواهیم کرد.

```
45 % Hala controller dovom ro taarif mikonim, chon mikaym 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 % Aval Bode plot ro mizarem baraye system ba controller
51 figure;
52 subplot(2, 2, 1);
53 bode(secondloopg);
54 grid on;
55 title('System with 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 with 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 with Controller - Step Response');
69 stepInfo2 = stepinfo(closedLoopSys2);
70 disp('System with 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 with Controller - Pole-Zero Map');
78
```

شکل ۴-۶

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



شکل ۵-۶

در نمودار اول میبینید که دامنه به اندازه ۶۰ دسیبل در فرکانس های پایین 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 ۰/۹۹ به ۰/۱ کاهش یافته است. همچنین با مقایسه دو شکل ۵-۶ و ۴-۶ متوجه کاهش نوسانات می‌شویم که ما را به پایداری نزدیک تر نیز می‌کند. با بالا آمدن نمودار دامنه این نمودار به مقدار ۰ دسیبل نیز میرسد این به معنی توانایی محاسبه حاشیه فاز است. با داشتن حاشیه فاز می‌توانیم پایداری این سیستم را بررسی کنیم همچنان حاشیه بهره بینهایت است.

System with Controller - Step Response Info:

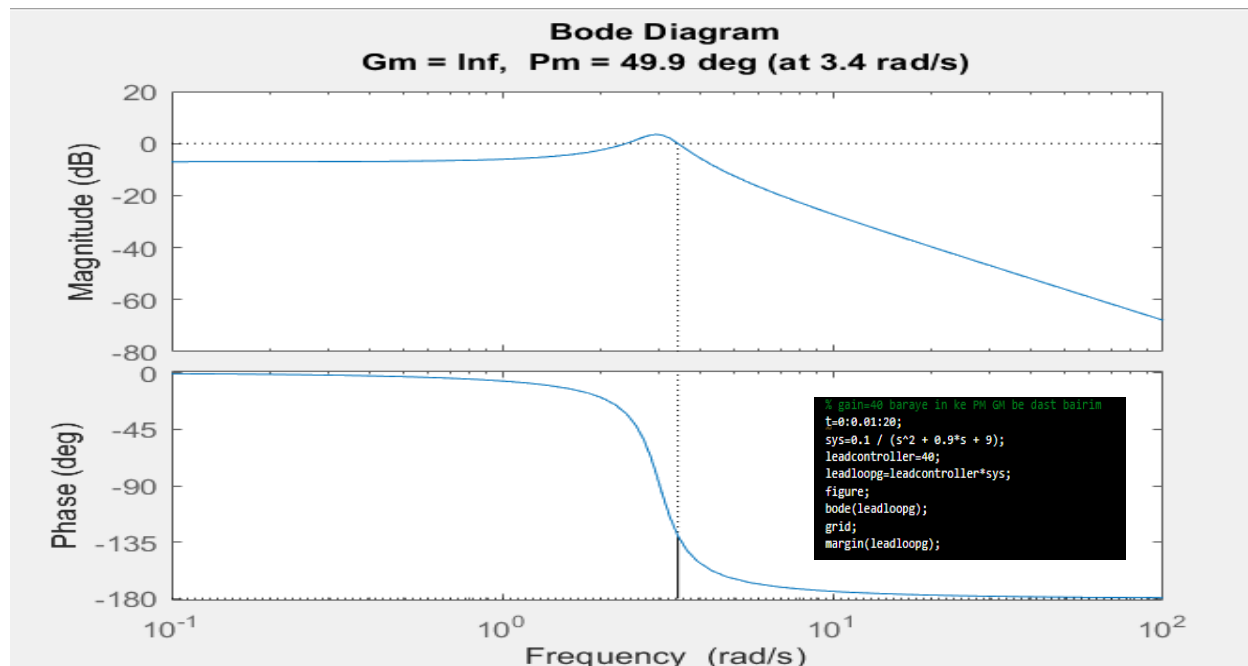
```
RiseTime: 0.0852
TransientTime: 0.6196
SettlingTime: 0.6196
SettlingMin: 0.8100
SettlingMax: 1.0229
Overshoot: 13.8483
Undershoot: 0
Peak: 1.0229
PeakTime: 0.2850
```

شکل ۶-۶

همانطور که مشاهده می‌شود، تمامی خواسته‌های کنترلی به خوبی برآورده شده است. زمان نشست سیستم به‌طور قابل توجهی کمتر از ۱۰ ثانیه است، که انتظار می‌رفت، چرا که استفاده از دو کنترلر Lead به میزان زیادی سرعت سیستم را افزایش داده است. این موضوع را می‌توانید در پاسخ زمانی شکل ۶-۵ و ۶-۲ نیز مشاهده کنید که سیستم دارای کنترلر با شیب خیلی زیادی به شمت ۱ می‌رود و به عبارتی سرعت این سیستم بهبود یافته‌است. همچنین، مقدار فرافروش در نبود کنترلر حدود ۶۰٪ است که به ۱۳٪ کاهش یافته است، این فرافروش در محدوده‌ای بین ۱۲٪ تا ۱۵٪ قرار دارد و این نشان‌دهنده این است که خواسته‌های کنترلی به‌خوبی محقق شده‌اند. توجه نمایید که به دلیل کاهش تایپ سیستم و حذف قطب در مبدا خطای ماندگار به ورودی پله هرگز صفر نمی‌شود 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 و وجود محدودیت های فیزیکی ممکن است به اندازی کافی بزرگ نشود مصالحه بین خواسته های کنترلی اجتناب ناپذیر است.

حال به بررسی حل دستی می‌پردازیم:

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 ۶-۷

از کنترل کننده lead به منظور کنترل فرکانس، زمان نشست استفاده می‌کنیم:

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

برای نمودار بودی در مرتب در دهانه کند 3.4 rad/s داریم حاشیه فاز ۴۹.۹ درجه

$$\phi = -\omega\tau + \theta = 0.12 \rightarrow \theta = 55.98^\circ$$

$$\phi = 55.98^\circ = 56^\circ \rightarrow P.M = \phi \times 100 = 55.98^\circ$$

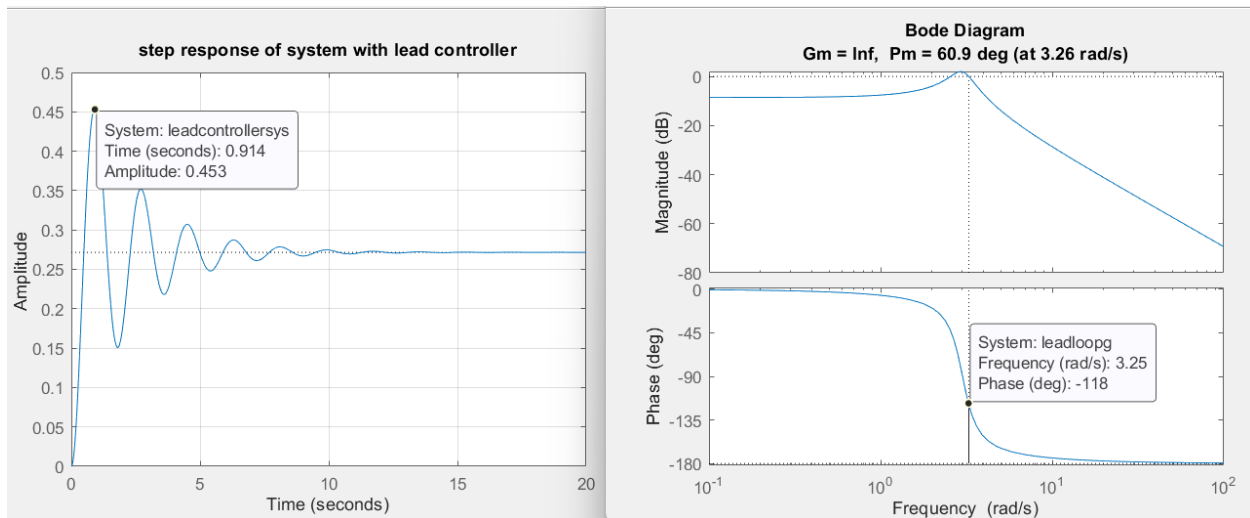
$$\varphi_m = 55.98 - 49.9 + 5 = 10.1^\circ$$

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

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

شکل ۶-۸

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



شکل ۹-۶

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

```
%نمودار های اضافه کردن کنترل گر 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)
```

Handwritten notes and calculations:

این فرجهش زیاد را با استفاده از کنترل گر lag حذف می‌کنیم.

$\xi = 0.05$

$k_c = 1.6$

$k_1 = k_c - 1 = 0.6$

$a = \frac{1}{k_c} = 0.625$

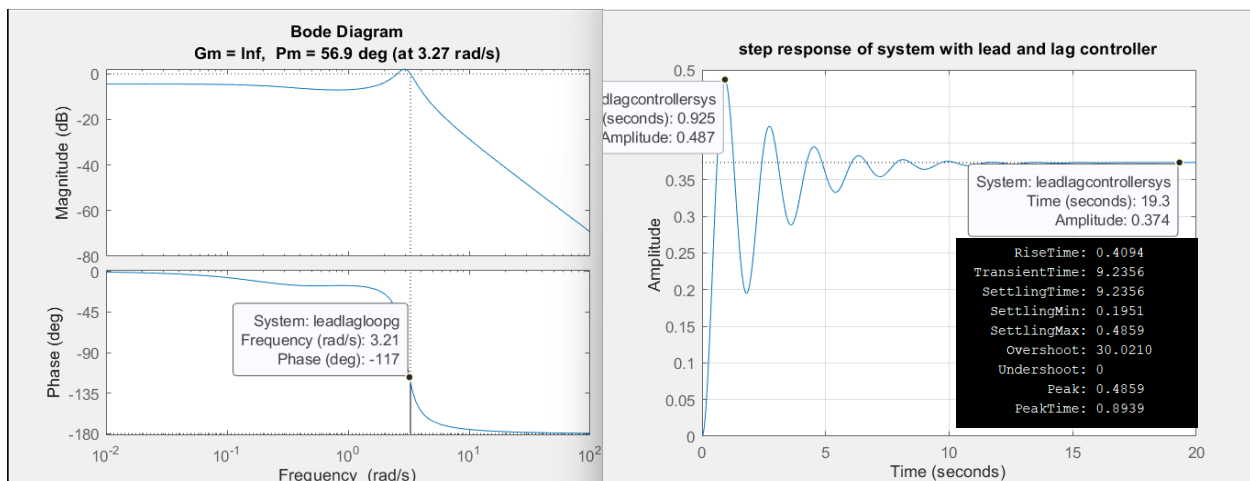
$C_{lag} = 1.6 \times \left(\frac{1.94s + 1}{3.1s + 1} \right)$

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

کنترل گر نهایی: $C = C_1 \times C_2$

شکل ۱۰-۶

نمودار بودی حلقه بسته نهایی با حاشیه فاز مناسب و پاسخ پله سیستم نهایی با فرجهش 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 ارضا کنند. برای سیستم تخمین زده شده کنترل کننده ای طراحی کنید که خطا ماندگار در تعقیب شیب ورودی را به زیر ۲ درصد برساند. با استفاده از تابع تبدیل حساسیت کنترل کننده ای طراحی کنید که فروجهش 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.

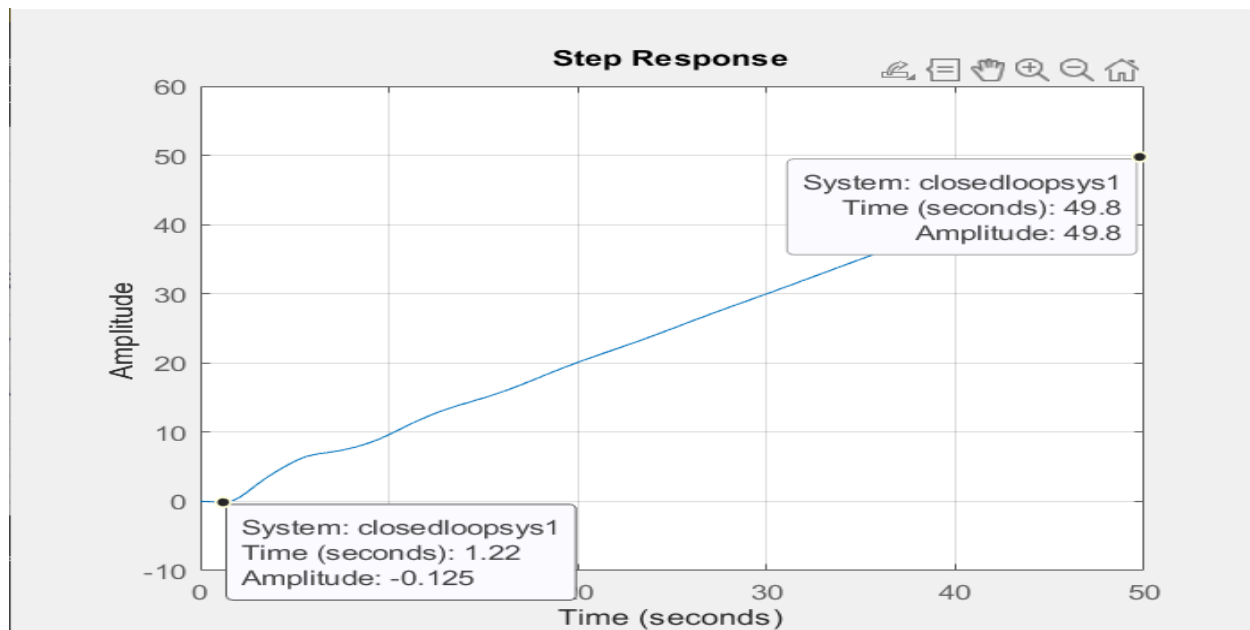
$$\lim_{s \rightarrow 0} S(s) = \lim_{s \rightarrow 0} \frac{S(0.18 - 0.2)}{S(s^2 + 0.9s + 9)} = -2/90$$

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

خطای حالت ماندگار منفی می‌باشد، نمودار به سمت منفی‌ها می‌رود. اول باید کسین منفی به هم تا نمودار بالا بیاید. برای اینکه خطا را به زیر ۲٪ برسانیم باید کسین را برای به هم. این کسین را می‌توان به هم تبدیل کرد چون می‌توانست نامیده می‌شود پس باید در عدد Lag قرار دهیم. این کار توسط Sisdeval، در ۵ p2 انجام شد، طبق نمودار در زمان ۴۵ تا ۵۰، پاسخ ورودی سیگنال دارای خطای ماندگار منفی است.

شکل ۷-۱

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



شکل ۷-۲

برای از بین بردن خطای ماندگار می‌توانیم گین را افزایشش دهیم مشکل اینجا است 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 PIDtuning به جای گین بالا از کنترلگر صنعتی pi استفاده می‌کنیم که باعث کاهش خطای ماندگار می‌شود. همانطور که در شکل ۷-۲ مشهود است ابتدا سیستم دارای خطا است اما با گذشت زمان این خطا به زیر ۲٪ کاهش می‌یابد.

استفاده از تابع حساسیت:

تابع حساسیت دارای یک هنر غیر کمینه فاز هست که این هنر در زمان ۱/۵۰۰ و ۲/۵۰۰ مکرر دارد.

از قیاس شکل آبی استفاده کنیم، می‌بینیم که هنر در زمان ۱/۵۰۰ مکرر دارد. چون این به بی‌ثباتی ۲ است

$$T_d = \frac{\frac{s_c}{c} + 1}{(s+1)^2}$$

امت زیر ۳ تقابل یک هنر داریم =

$$T_d(2) = 0 \Rightarrow c = -2$$

بنابراین تابع تبدیل متعین حساسیت می‌باشد:

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

کنترلر نهایی:

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

شکل ۷-۳

با استفاده از ابزار sisotool در مطلب تلاش می‌کنیم گینی را پیدا کنیم که در آن خواسته‌های کنترلی سوال به درستی برقرار شود، توجه نمایید که سیستم ما دارای یک صفر غیر کمینه 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 nakhsh 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);

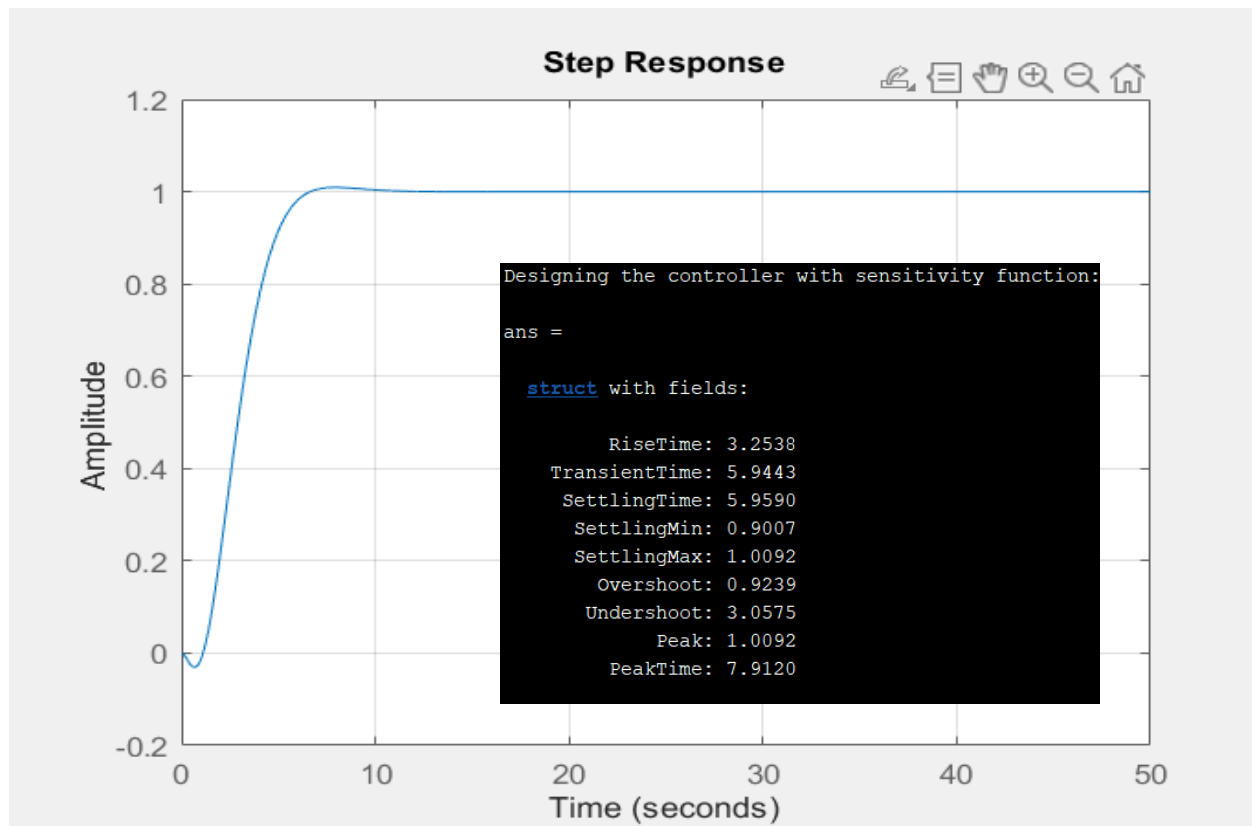
secondloopg = secondcontroller * sys;
closedloopsys2 = secondloopg / (1 + secondloopg);

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
```

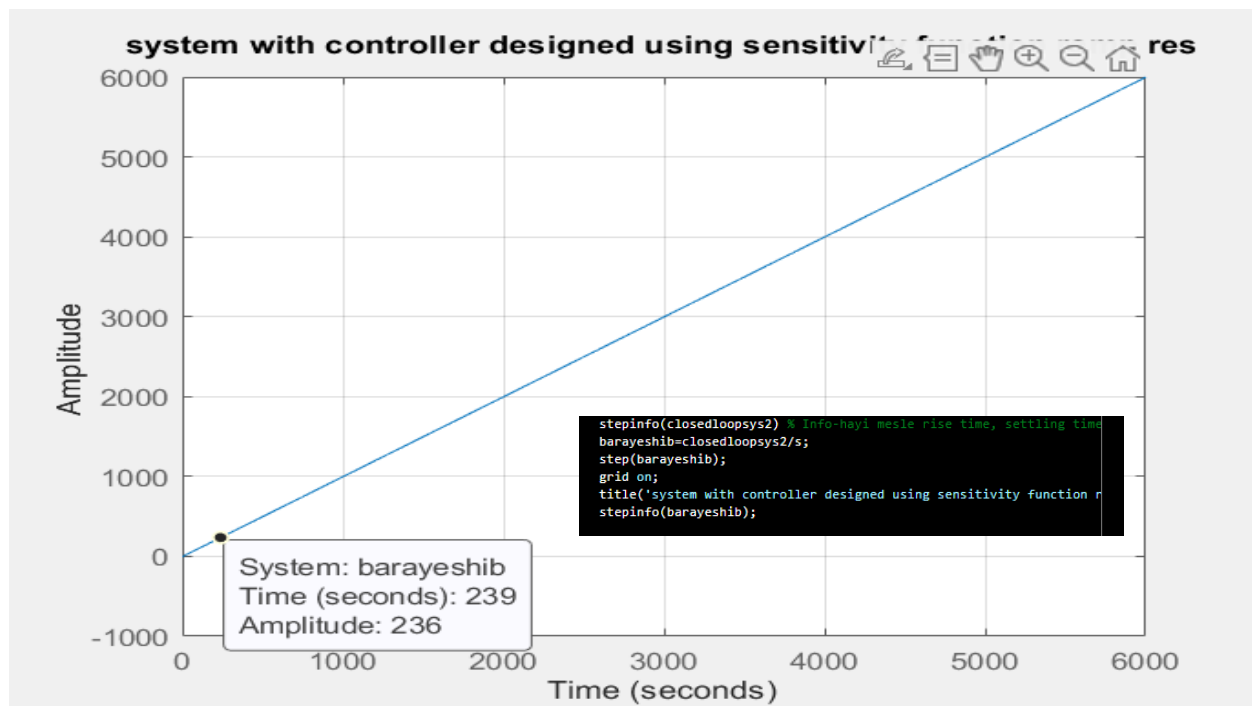
شکل ۷-۴

خروجی این کد در شکل بعدی آورده شده خواهید 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.



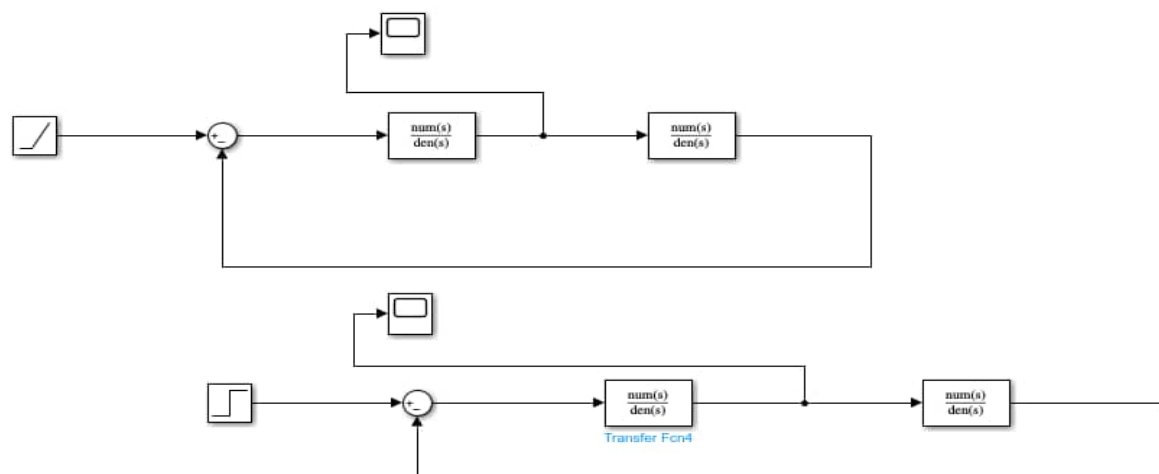
شکل ۵-۷

این سیستم دارای فروجهش کمتر از ۶٪ و زمان نشست زیر ۶ ثانیه است. همان طور که انتظار میرفت خطا به ورودی پله ۰ است و همچنین ورودی شیب را به خوبی تعقیب می‌کند. توجه نمایید به دلیل درصد کم خطا متلب آن را نمایش نداد اما مقدار خیلی کمی خطا دارد.

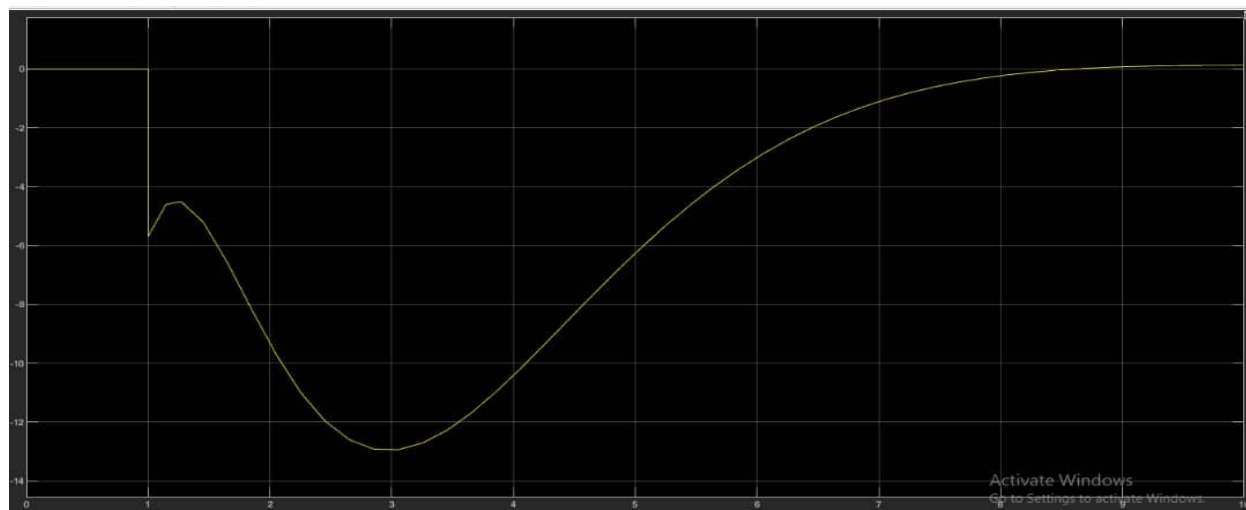


شکل ۶-۷

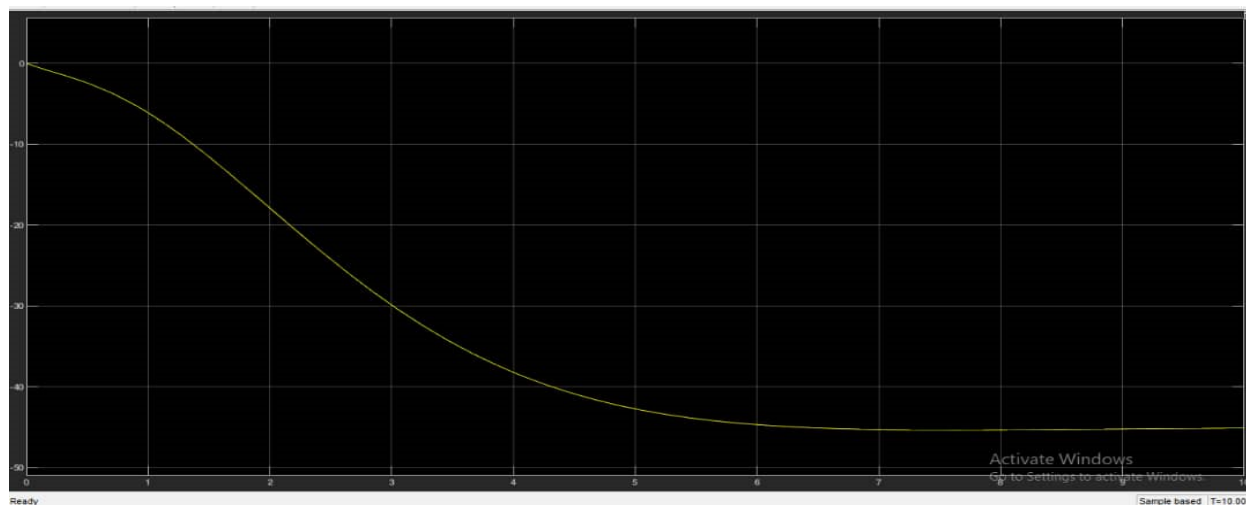
در ادامه با استفاده از Simulink نرم افزار متلب می توانیم برای هر دو ورودی پله و شیب تلاش کنترلی را بررسی کنیم . در ادامه بلوک دیگر طراحی شده به این منظور و نمودار مربوط به control effort خدمتان ارسال می کنیم.



شکل ۷-۷



شکل ۷-۸



شکل ۷-۹

نمودار های موجود در شکل ۸-۷ و ۹-۷ مربوط به خروجی بلوک 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 خطا و به عبارتی تلاش کنترلی ای نداریم اما در در نمودار دوم می‌بینید که برای ورودی شیب همانطور که انتظار می‌رفت نیاز به تلاش کنترلی داریم چرا که سیستم های تایپ ۱ به ورودی شیب خطا 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 شیب 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.