بسم الله الرحمن الرحیم

گزارش کار آزمایش شماره 3 - آزمایشگاه کنترل خطی

اعضای گروه: سیده لیلا حسینی **9923024** مریم مقتدری **9923073** زهرا لطیفی **9923069**

نام آزمایش: کنترل روبات با مفصل نرم (Robot Joint Flexible)

هدف: کنترل ربات با استفاده از جبران ساز Lag و Lead و همچنین روش فیدبک حالت

هدف از انجام آزمایش‌های این بخش، به کارگیری تکنیک‌های کنترل آموخته شده برای کنترل رفتار بازو با مفصل نرم وکاهش نوسانات انتهای بازو می‌باشد. پس از آن در قسمت‌های بعدی به مطالعه پاسخ فرکانسی این سیستم به ورودی با فرکانس‌های مختلف و تاثیر تاخیر در رفتار این سیستم خواهیم پرداخت. سیستم مورد آزمایش در آزمایشگاه متشکل از یک سروو موتور است که یک مفصل نرم را می‌گرداند. اتصال بین بازوی سخت سوار شده روی مفصل و خود بدنه مفصل از طریق دو عدد فنر و یک نقطه اتصال می‌باشد تا بتوان ساختاری انعطاف‌پذیر را به صورت کامل محقق ساخت. کنترل این سیستم مشابه مسائل کنترلی در ربات‌هایی با مقیاس‌های خیلی بزرگتر در صنعت می‌باشد که در مفاصل آن ها انعطاف رخ می‌دهد.

• کنترل کننده فیدبک حالت

در این بخش طراحی یک کنترل کننده فیدبک حالت را با نوشتن معادلات دینامیکی یک مفصل منعطف بررسی کردیم. تابع تبدیل این سیستم به کمک فضای حالت و معادلات دینامیکی آن که از روش اویلر-لاگرانژ بدست آمده‌اند، حاصل شد. ابتدا با پاسخگویی به سوالات دستورکار، مروری بر معادلات دینامیکی مربوط به مفصل منعطف داریم:

سوال: معادله نهایی توصیف کننده K_{Stiff} را بدست آورید.

$$K_{Stiff} = -RF_y$$

سوال: با فرض در دسترس بودن ممان چرخشی بدنه و همچنین ممان چرخشی بازو، انرژی پتانسیل و جنبشی کلی سیستم را محاسبه نمایید.

$$K_{Total} = \frac{1}{2}J_{hub}\dot{\theta}^2 + \frac{1}{2}J_{arm}\left(\dot{\theta} + \dot{\alpha}\right)^2$$

$$P = \frac{1}{2}K_{Stiff}\alpha^2$$

سوال: ترم$\boldsymbol{B_{eq}\dot{\theta}}$ بیانگر چه پارامتری در سیستم می‌باشد؟

ترم $B_{eq}\dot{\theta}$ نمایانگر اثر اصطکاک می‌باشد.

سوال: با استفاده از معادالت دینامیکی نهایی سیستم و با در نظر گرفتن فرم فضای حالت زیر به صورت زیر ماتریسهای توصیف فضای حالت برای سیستم مفصل نرم را به دست آورید (خروجی سیستم شامل هر دو حالت θ و α باشد.)

$$X=\begin{bmatrix}\theta\\ \alpha\\ \dot{\theta}\\ \dot{\alpha}\end{bmatrix}\rightarrow \dot{X}=\begin{bmatrix}\dot{\theta}\\ \dot{\alpha}\\ \ddot{\theta}\\ \ddot{\alpha}\end{bmatrix}\xrightarrow[Y=CX+DU]{\dot{X}=AX+BU}$$

$$A=\begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & \dfrac{K_{Stiff}}{J_{eq}} & \dfrac{\eta_m\eta_g K_t {K_g}^2 K_m+B_{eq}R_m}{J_{eq}R_m} & 0\\ 0 & -\dfrac{K_{Stiff}\left(J_{eq}+J_{Arm}\right)}{J_{eq}J_{Arm}} & \dfrac{\eta_m\eta_g K_t {K_g}^2 K_m+B_{eq}}{J_{eq}R_m} & 0\end{bmatrix}, B=\begin{bmatrix}0\\ 0\\ \dfrac{\eta_m\eta_g K_t K_g}{J_{eq}R_m}\\ -\dfrac{\eta_m\eta_g K_t K_g}{J_{eq}R_m}\end{bmatrix}$$

$$C=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix} \quad , \quad D=\begin{bmatrix}0\\ 0\end{bmatrix}$$

با استفاده از مقادیر موجود در دستورکار:

$$A=\begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 480.13 & -28.26 & 0\\ 0 & -837.21 & 28.26 & 0\end{bmatrix}, B=\begin{bmatrix}0\\ 0\\ 49.7\\ -49.7\end{bmatrix}$$

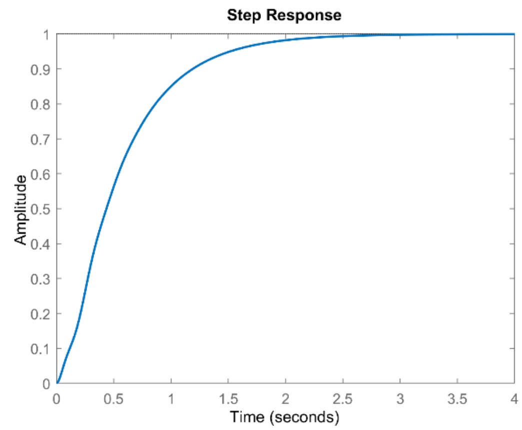
$$C=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix}, D=\begin{bmatrix}0\\ 0\end{bmatrix}$$

سوال: پس از یافتن فضای حالت، تابع تبدیل $\frac{\theta(s)}{V_m(s)}$ و $\frac{\alpha(s)}{V_m(s)}$ را بدست آورده و پاسخ ضربه و پله هریک از این توابع تبدیل را رسم نمایید.

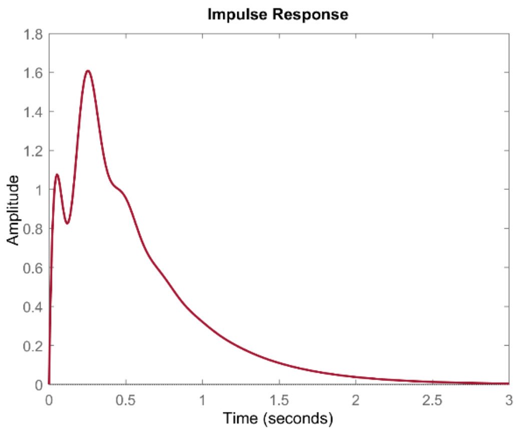
با توجه به رابطه ماتریس تبدیل حالت، تابع تبدیل‌های سیستم را به دست می‌آوریم:

$$G_1(s)=\frac{\theta(s)}{V_m(s)}=C(sI-A)^{-1}B+D=\frac{49.7s^2+17600}{s^4+28.26s^3+837.2s^2+10010s}$$

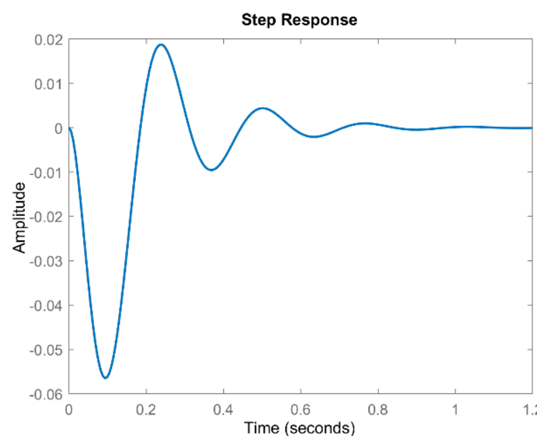
$$G_2(s) = \frac{\alpha(s)}{V_m(s)} = C(sI - A)^{-1}B + D = \frac{-49.7s^2}{s^4 + 28.26s^3 + 837.2s^2 + 10010s}$$



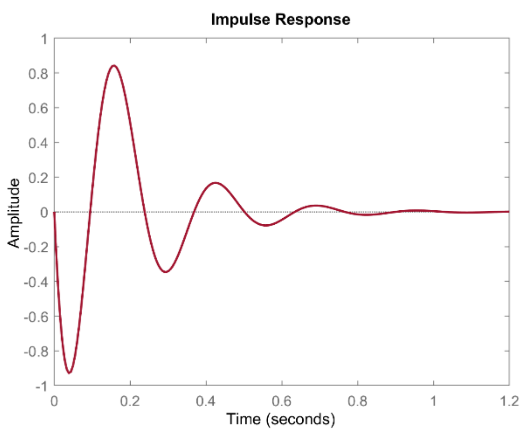
1) پاسخ پله سیستم حلقه بسته G1



2) پاسخ ضربه سیستم حلقه بسته G1



3) پاسخ پله سیستم حلقه بسته G2



4) پاسخ ضربه سیستم حلقه بسته G2

دو تابع تبدیل بالا با کد زیر و نمودارها هم با دستور step و impulse در متلب حاصل شده‌اند:

```
%Transfer Function
Statespace = ss(A, B, C, D);
display(Statespace);

[num_G, den_G] =ss2tf(A, B, C, D);
G_1 = tf(num_G(1, :), den_G);
G_2 = tf(num_G(2, :), den_G);
G = [G_1; G_2];
display(G);
```

```
Command Window

G =

  From input to output...
             49.7 s^2 + 5.215e-27 s + 1.76e04
   1:  ------------------------------------------
       s^4 + 28.26 s^3 + 837.2 s^2 + 1.001e04 s

         -49.7 s^2 + 2.497e-13 s + 1.998e-12
   2:  ------------------------------------------
       s^4 + 28.26 s^3 + 837.2 s^2 + 1.001e04 s

Continuous-time transfer function.
```

درباره قطب‌های سیستم هم می‌توان گفت همان مقادیر ویژه ماتریس A خواهند بود.

حال به سراغ کنترل کننده می‌رویم. به کمک فیدبک حالت می‌توان قطب‌های سیستم را جابجا کرد تا شرایط مطلوب فراهم شود. قبل از استفاده از این روش باید کنترل‌پذیری سیستم را بررسی کنیم. کنترل‌پذیری یک سیستم را می‌توان به کمک رنک ماتریس کنترل‌پذیری بررسی کرد. اگر رنک این ماتریس برابر تعداد حالت‌های سیستم باشد، سیستم را کنترل‌پذیر می‌گوییم. (یعنی با ورودی‌های سیستم می‌توان خروجی را کنترل کرد.)

$$C_{AB} = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B], \qquad rank(C_{AB}) = n$$

ماتریس Controllability را تشکیل می‌دهیم:

$$C_{AB} = [B \quad AB \quad A^2B \quad A^3B] = \begin{bmatrix} 0 & 49.7 & -1404.5 & 15680 \\ 0 & -49.7 & 1404.5 & 1917.5 \\ 49.7 & -1404.5 & 15680 & 235440 \\ -49.7 & 14045 & 1917.5 & -732760 \end{bmatrix} \rightarrow rank(C_{AB}) = 4$$

می‌بینیم که سیستم کنترل‌پذیر است. کنترلر حالت ورودی سیستم را تغییر می‌دهد و فرم کلی آن (برای این سیستم) به صورت زیر می‌باشد:

$$u = r - Kx(t), \qquad K = [K_\theta \quad K_\alpha \quad K_{\dot\theta} \quad K_{\dot\alpha}] \rightarrow$$
$$\dot{x} = (A - BK)x(t)$$

گفتیم قطب‌های سیستم حلقه بسته مقادیر ویژه ماتریس A است. پس قطب‌های سیستم جبرانسازی شده به صورت زیر می‌باشد:

$$\varDelta(s) = \det(sI - A + BK) = 0 \Rightarrow (s-\lambda_1)(s-\lambda_2)(s-\lambda_3)(s-\lambda_4) = 0$$

اما قطب‌های مطلوب به شکل زیر به ما داده شده بود:

```
sd = [-5.04+1j*5.141 -5.04-1j*5.141 -100 -50];
```

با کمک دو رابطه بالا ماتریس K قابل محاسبه است. روش دیگر برای محاسبه این ماتریس‌ها، استفاده از دستور place است. این تابع دو ماتریس A و B و محل قطب‌های مطلوب را به عنوان ورودی می‌گیرد و خروجی آن بردار K می‌باشد:

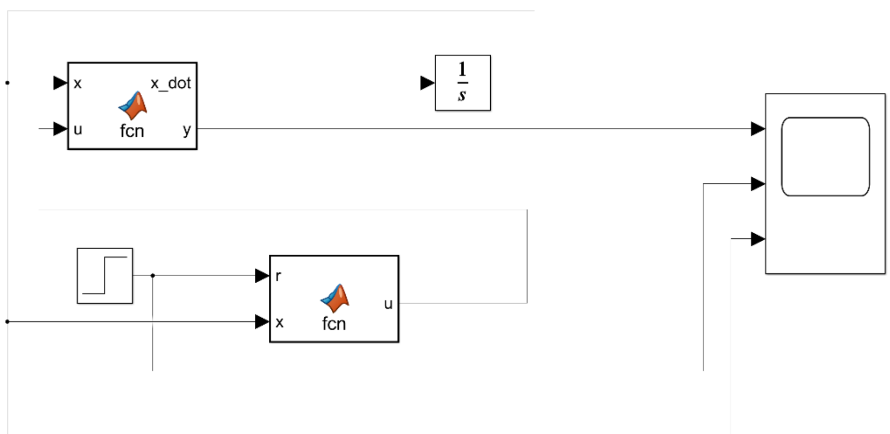
```
C_c = ctrb(A, B);
r = rank(C_c);
if r == length(A)
    sd = [-5.04+1j*5.141 -5.04-1j*5.141 -100 -50];
    k = place(A, B, sd);
    display(k);
else
    disp("It's not CONTROLLABLE!!!");
```

اگر فقط قسمت‌های حقیقی قطب‌ها را در نظر بگیریم، می‌توان از دستور acker هم استفاده کرد.

بردار K به این صورت برای ما محاسبه شد:

$$K = [14.7 \quad -100.49 \quad 2.7 \quad 0.0848]$$

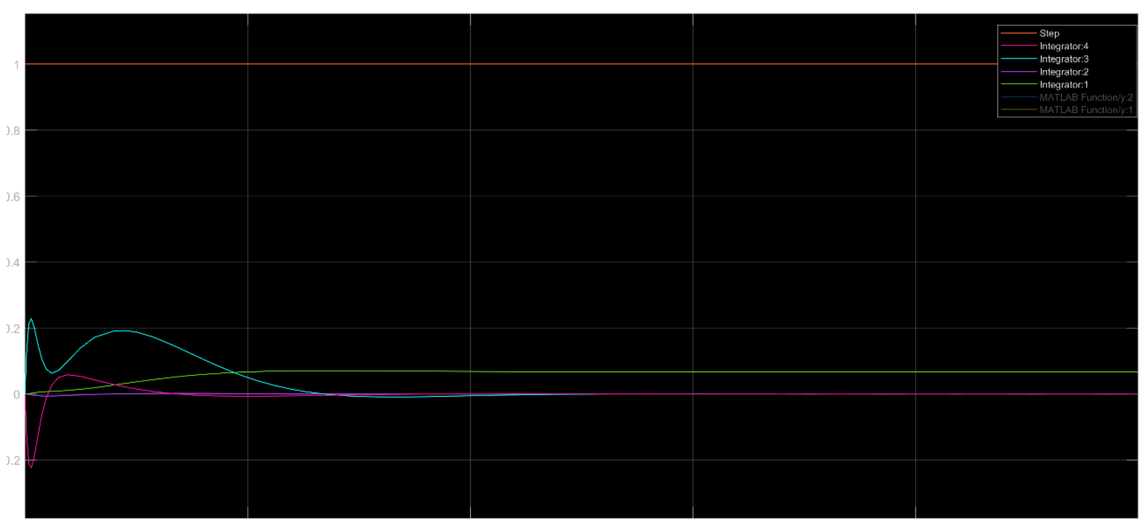
پس از محاسبه این بردار، در محیط Simulink یک MATLAB Function تعریف می‌کنیم که دو ورودی x و u دارد و خروجی آن $\dot{x}$ و y می‌باشند. (توابع موجود در این بلوک همان $\dot{x} = Ax + Bu$ و $y = Cx + Du$ می‌باشند.) ورودی u این تابع را با یک MATLAB Function دیگر، r – Kx قرار می‌دهیم. از خروجی سیستم به کمک یک انتگرال‌گیر ($\frac{1}{s}$)، انتگرال می‌گیریم و x را ایجاد می‌کنیم:



5) استفاده از MATLAB Function در محیط Simulink

باید توجه داشت که چون درحال کار با ماتریس‌ها هستیم، باید شرایط اولیه انتگرال‌گیر را به این صورت داد:

```
zeros(4,1)
```

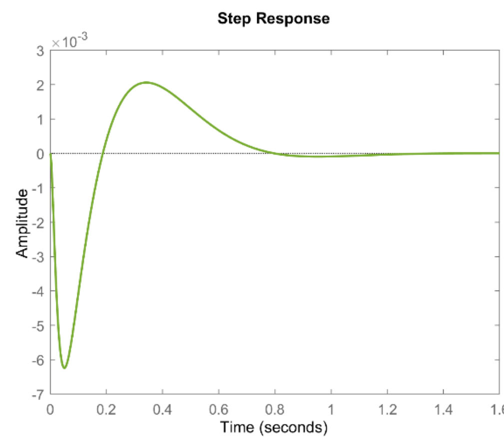


6) خروجی Scope به ورودی پله

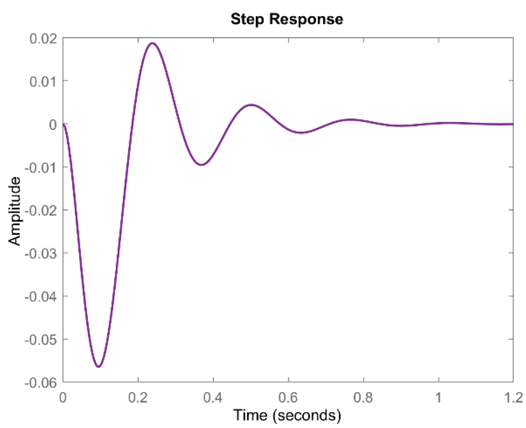
اسکوپ ما 3 پورت ورودی دارد که رنگ نارنجی ورودی پله را نمایش می‌دهد، سبز و بنفش y یا خروجی متلب فانکشن را و آبی و صورتی بردار x را. باتوجه به ماتریس C و D می‌دانیم که خروجی y با دو مولفه اول بردار x یکی است و همان آلفا و تتا را نمایش می‌دهد. اما دو مولفه آخر x، با مشتق آلفا و مشتق بتا برابر است.

- **نتیجه گیری**

1. پاسخ سیستم حلقه بسته (θ و α) را به ورودی پله رسم کنید و بیشینه انحراف α، زمان نشست، زمان صعود و درصد بالازدگی سیستم حلقه بسته را محاسبه نمایید.



7) پاسخ پله جبران شده α



8) پاسخ پله جبران شده θ

```
Command Window
     RiseTime: 0.3744
 SettlingTime: 0.8688
  SettlingMin: 0.0611
  SettlingMax: 0.0706
    Overshoot: 3.9572
   Undershoot: 0
         Peak: 0.0706
     PeakTime: 0.6724
fx
```

9) اطلاعات پاسخ پله α

```
Command Window
     RiseTime: 5.8287e-16
 SettlingTime: 0.7361
  SettlingMin: -9.4673e-05
  SettlingMax: 0.0021
    Overshoot: 8.9067e+15
   Undershoot: 2.6983e+16
         Peak: 0.0062
     PeakTime: 0.0507
fx
```

10) اطلاعات پاسخ پله θ

2. علت بروز تفاوت احتمالی بین پاسخ شبیه‌سازی و پاسخ عملی را توضیح دهید.

علت این اختلاف تقریب‌هایی است که در محاسبات استفاده می‌شوند و همچنین خطایی است که افزودن کنترل‌کننده‌ها به سیستم اضافه می‌کند. همچنین در پیاده‌سازی عملی عواملی همچون نویز و خطای فیزیکی دستگاه در اختلاف نسبت به مقدار تئوری دخیل است.

• کنترل کننده Lead

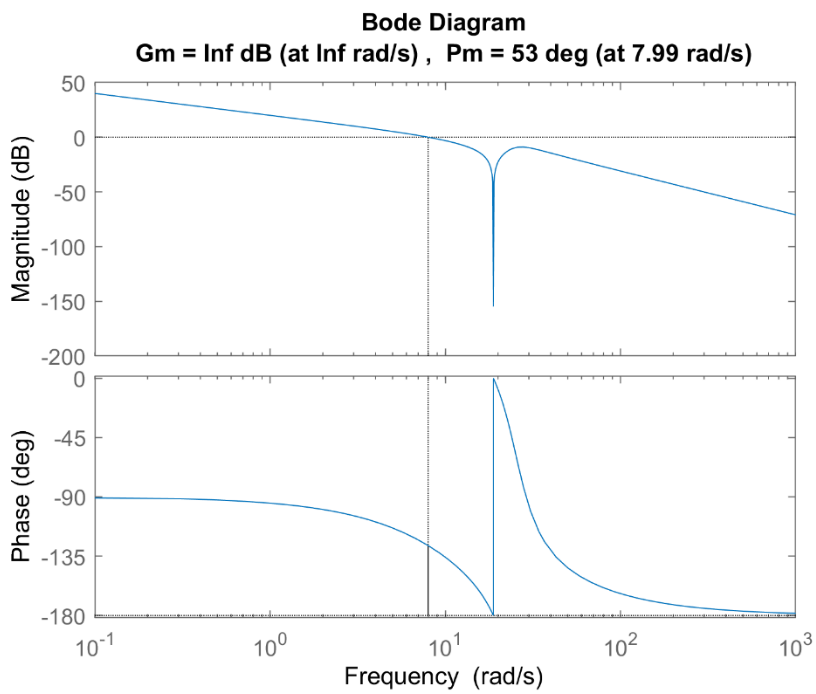
در این بخش به طراحی یک کنترل کننده Lead در حوزه فرکانس برای معادلات دینامیکی این مفصل منعطف می‌پردازیم. معادلات فضای حالت و در نتیجه آن توابع تبدیل، مشابه قسمت قبل هستند. در این قسمت باید کنترل کننده‌ای طراحی کنیم که پاسخ سیستم را به شرایط مطلوب ما نزدیک کند.

کنترلر lead به سیستم گین و فاز اضافه می‌کند. برای طراحی در حوزه فرکانس دو معیار حدفاز مطلوب و خطای k_v مطلوب در اختیار ما قرار گرفت.

$$\begin{cases} k_v = 10 = \lim\limits_{s\to 0} |sCG| \\ C(s) = k\alpha \dfrac{1+Ts}{1+\alpha Ts} \end{cases} \Rightarrow \frac{17600}{10010} k\alpha = 10 \Rightarrow k\alpha \cong 5.68$$

حال نمودار Bode سیستم $k\alpha G(s)$ را با دستور زیر رسم کرده و با توجه به حدفاز موجود و حد فاز مطلوب (70 درجه) مقدار φ_m را بدست می‌آوریم:

```
margin(ka*G_1);
```



11) رسم نمودار Bode و مشخص کردن Phase Margin تابع تبدیل kαG(s)

$$\varphi_m = \varphi_d - \varphi_{P_m} + \Delta \Rightarrow \varphi_m = 70 - 53 + 5 = 22°$$

مقدار Δ اضافه شده تا تاثیر افزایش گین در فرکانس و انتقال آن به مقادیر بالاتر را جبران کند.

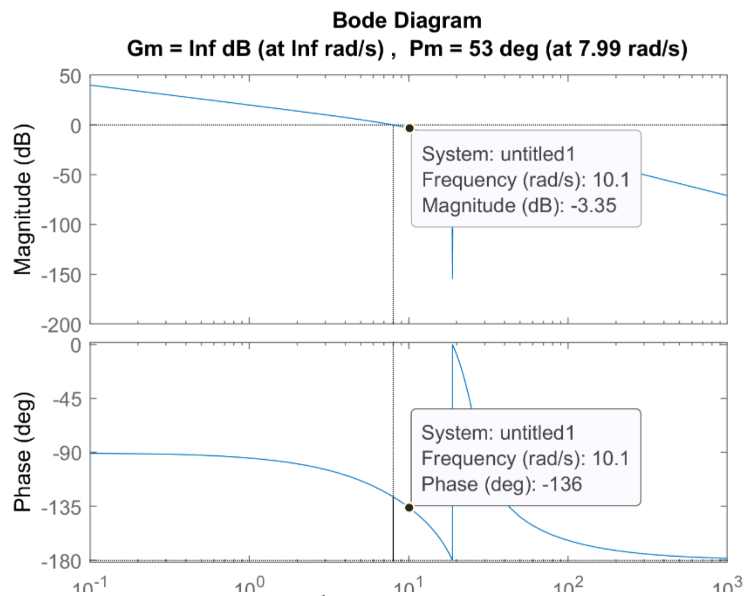
$$\sin\varphi_m = \frac{\alpha - 1}{\alpha + 1} \Rightarrow \alpha \cong 2.17 \xrightarrow{k\alpha \cong 5.68} k = 2.617$$

برای این فرکانس قطع جدید در ω_n قرار بگیرد باید اندازه تابع تبدیل معکوس اندازه کنترل‌کننده در این فرکانس باشد:

$$|C(j\omega_n)|_{dB} = 10\log\alpha \Rightarrow |G(j\omega_n)|_{dB} = -10\log\alpha \cong -3.36$$

با استفاده دیاگرام Bode مقدار ω_n را می‌یابیم که تقریبا برابر 10.1 رادیان بر ثانیه است.

$$\begin{cases} \alpha = \dfrac{Z}{P} \\ \omega_n = \sqrt{ZP} \end{cases} \Rightarrow P \cong 14.88, \quad Z \cong 6.85$$

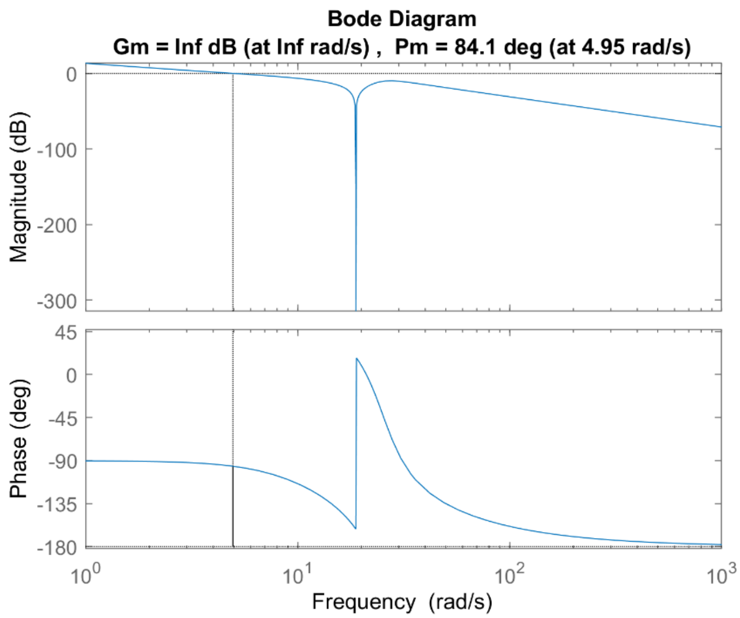


12) محاسبه ω_n از روی دیاگرام Bode

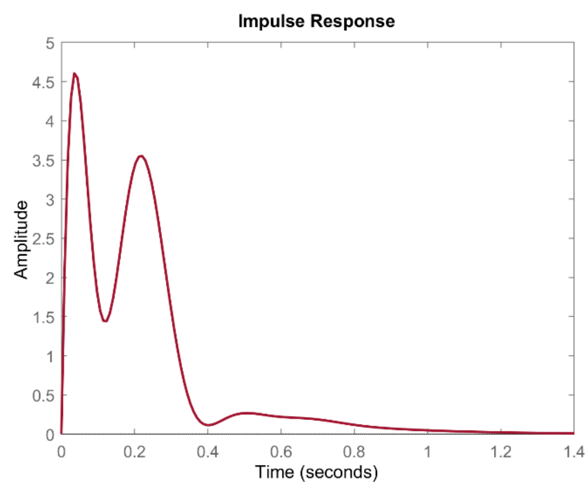
حال تابع تبدیل کنترل‌کننده حاصل شده و می‌توان سیستم جبران شده را با ضرب آن در تابع تبدیل جبران نشده، بدست آورد:

$$C(s) = \frac{s + 6.85}{s + 14.88}$$

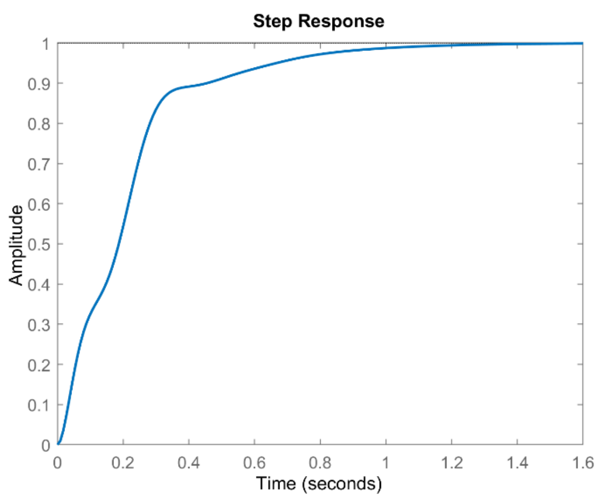
اکنون سیستم جبران شده را تشکیل داده، دیاگرام Bode آن را رسم کرده و پاسخ پله و ضربه سیستم حلقه بسته آن را مشاهده می‌کنیم.



13) دیاگرام Bode سیستم جبران شده



14) پاسخ ضربه سیستم حلقه بسته جبران شده



15) پاسخ پله سیستم حلقه بسته جبران شده

مشاهده می‌شود که حدفاز بدست آمده (84.1 درجه) از حدفاز مطلوب بیشتر است. این امر به دلیل دقیق نبودن مقدار دلتا در نظر گرفته شده می‌باشد. با چند مرحله تکرار با مقادیر مختلف برای دلتا و بدست آوردن مقادیر جدید Z و P می‌توان به مقدار مطلوب نزدیک‌تر شد.

- **نتیجه گیری**

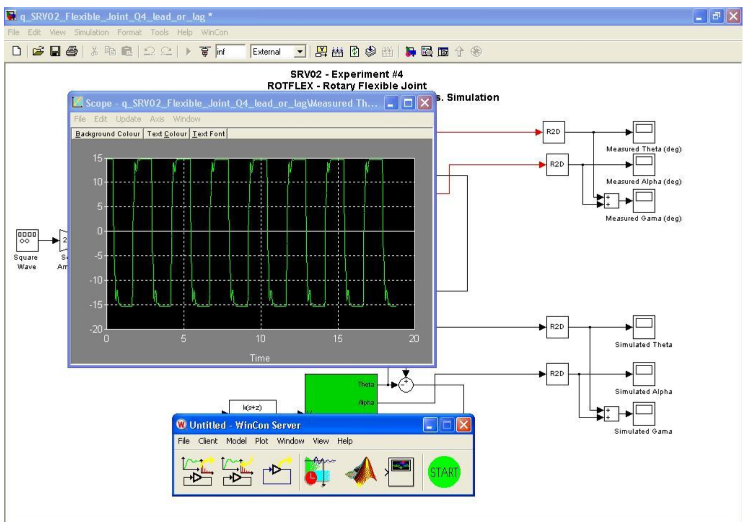
1. **پاسخ سیستم حلقه بسته (θ و α) را به ورودی پله رسم کنید و بیشینه انحراف α، زمان نشست، زمان صعود و درصد بالازدگی سیستم حلقه بسته را محاسبه نمایید.**

همانطور که در تصویر بالا مشاهده می‌شود، درصد بالازدگی صفر و زمان نشست (با معیار 2 درصد) تقریبا 0.88 ثانیه می‌باشد.

```
Command Window
    SettlingTime: 0.8831
     SettlingMin: 0.9007
     SettlingMax: 0.9993
       Overshoot: 0
      Undershoot: 0
            Peak: 0.9993
        PeakTime: 1.7534

fx >>
```

2. **علت بروز تفاوت احتمالی بین پاسخ شبیه‌سازی و پاسخ عملی را توضیح دهید.**



16) نتیجه شبیه‌سازی عملی سیستم جبران شده با کنترلر Lead

علت این اختلاف تقریب‌هایی است که در محاسبات استفاده می‌شوند و همچنین خطایی است که افزودن کنترل‌کننده‌ها به سیستم اضافه می‌کند. همچنین در پیاده‌سازی عملی عواملی همچون نویز و خطای فیزیکی دستگاه در اختلاف نسبت به مقدار تئوری دخیل است.

3. **کنترل کننده Lead چه مزایایی نسبت به کنترل کننده PD دارد؟**

تحقق فیزیکی PD سخت و همچنین به دلیل وجود یک مشتق‌گیر، باعث ورود نویز به سیستم و تقویت آن می‌شود. به همین دلیل از Lead استفاده می‌کنیم.

• کنترل کننده Lag

از دستورکار می‌دانیم از دلایل استفاده از کنترلرPI کاهش خطای حالت دائم و افزایش ثوابت خطا می باشد و کنترلر Lag تقریبی از PI است به طوری که توسط المان های پسیو قابل تحقق است. به طور کلی هدف از طراحی جبرانساز Lag در حوزه فرکانس رسیدن به حالت دائمی و حاشیه فاز مطلوب می باشد در عین حال که شرایط گذرا را دستخوش تغییر ننماید. جبرانساز Lag دارای مشخصه ی پایین گذر بوده به طوری که قطب آن نسبت به صفرش به محور $j\omega$ نزردیکتر است.

در این بخش به طراحی یک کنترل کننده Lag در حوزه فرکانس برای معادلات دینامیکی این مفصل منعطف می‌پردازیم. معادلات فضای حالت و در نتیجه آن توابع تبدیل، مشابه قسمت قبل هستند. در این قسمت باید کنترل کننده‌ای طراحی کنیم که پاسخ سیستم را به شرایط مطلوب ما نزدیک کند. این طراحی به طور کلی دارای مراحل زیر است:

تنظیم ثابت خطا (k) ، بدست آوردن ω'_c از روی حاشیه فاز، قرار دادن $Zc = 0.1\omega'_c$، محاسبه α از روی این رابطه: $|G(j\omega'_c)|_{dB} + 20\log\alpha = 0$، محاسبه Pc از روی رابطه مقابل: $Zc = \alpha Pc$.

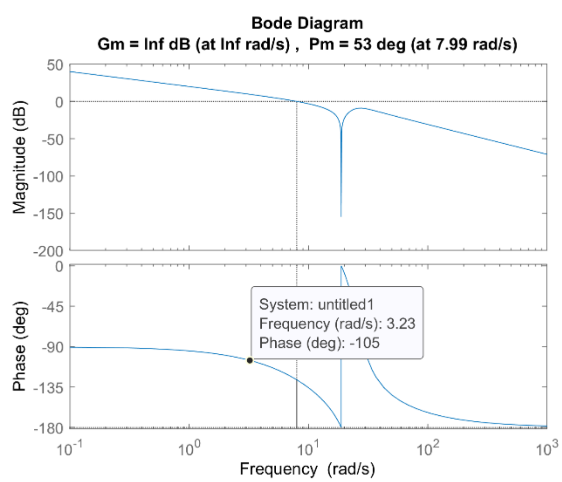
کنترل کننده lag با کاهش ω_c باعث بهبود حدفاز و کاهش پهنای باند می‌شود و کاهش پهنای باند یک نتیجه منفی و یک نتیجه مثبت را به دنبال دارد. حذف نویز و کاهش سرعت پاسخ، این نتایج هستند.

طبق دستورکار کنترل کننده Lag ما باید همان ویژگی‌های بخش قبل را داشته باشد. $\Rightarrow k \cong 5.68$

برای اطمینان از پاسخ، PM را 5 الی 10 درجه بیشتر از 70 که مطلوب ماست، درنظر می‌گیریم:

$$PM = 75° \rightarrow \measuredangle G(j\omega c) = -105°$$

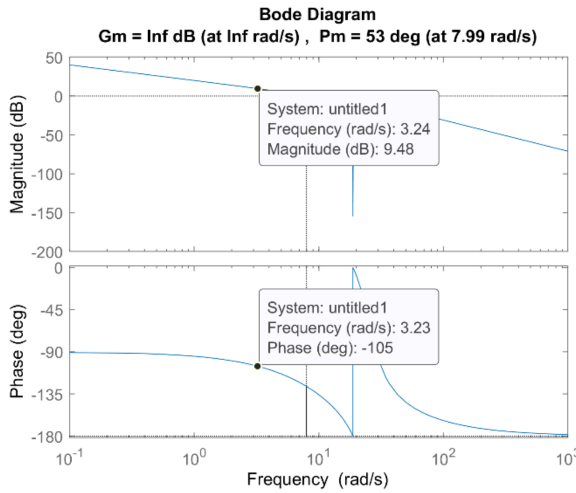
بار دیگر با استفاده از دستور Margin متلب، فرکانس متناظر با فاز -105° را می‌یابیم.



17) دیاگرام Bode سیستم جبران نشده

مشاهده شد که $\omega'_c = 3.23\frac{rad}{s}$ پس خواهیم داشت: $Zc = 0.1\omega'_c = 0.323$

حال دیگر بار به دیاگرام Bode رسم شده بازگشته و اندازه متناظر با فرکانس ω'_c را بدست می‌آوریم:



18) بدست آوردن اندازه متناظر از روی دیاگرام Bode

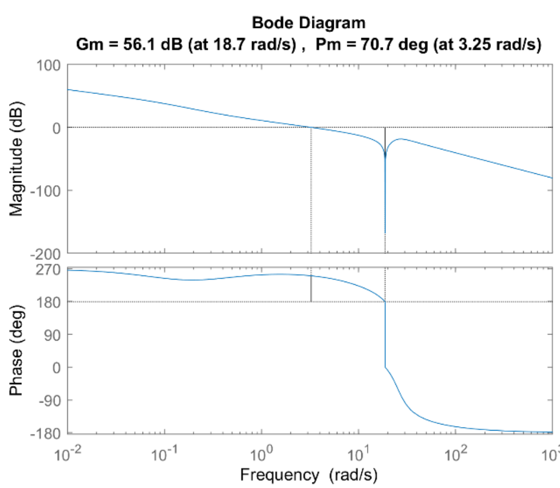
مشاهده می‌شود که اندازه تقریبی متناظر با ω'_c، برابر با 9.48 است. پس خواهیم داشت:

$$20\log\alpha + 9.48 = 0 \ \rightarrow\ \alpha \cong 0.336 \xrightarrow{Zc=\alpha Pc} Pc = 0.108$$

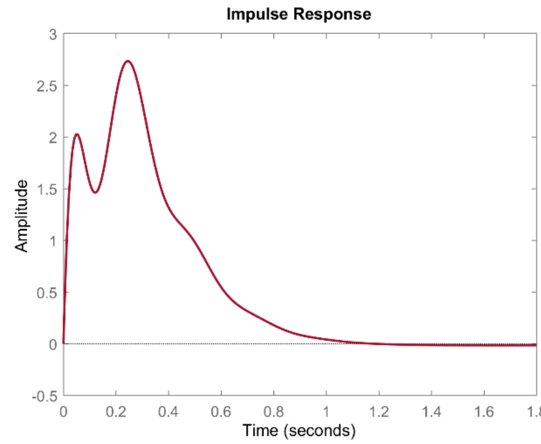
حال تابع تبدیل کنترل‌کننده حاصل شده و می‌توان سیستم جبران شده را با ضرب آن در تابع تبدیل جبران نشده، بدست آورد:

$$C(s) = 0.336 \times \frac{s+0.323}{s+0.108}$$

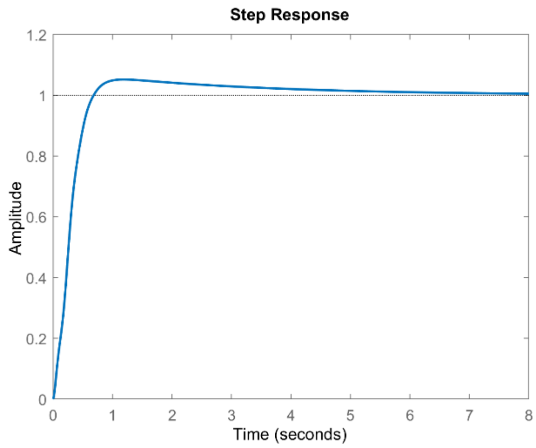
اکنون سیستم جبران شده را تشکیل داده، دیاگرام Bode آن را رسم کرده و پاسخ پله و ضربه سیستم حلقه بسته آن را مشاهده می‌کنیم.



19) دیاگرام Bode سیستم جبران شده



20) پاسخ ضربه سیستم حلقه بسته جبران شده

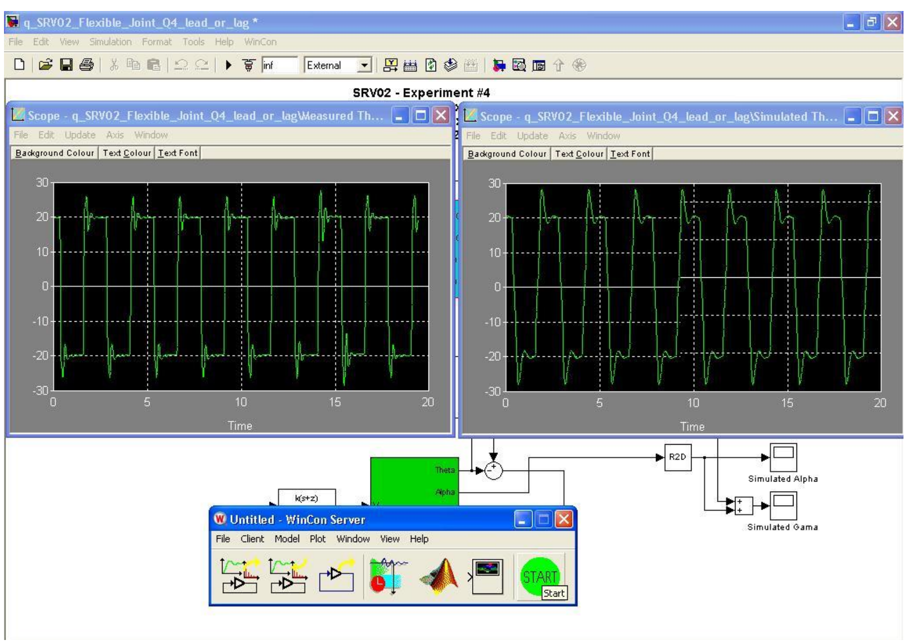


21) پاسخ پله سیستم حلقه بسته جبران شده

اطلاعات پاسخ پله سیستم جبران شده به شرح زیر است:

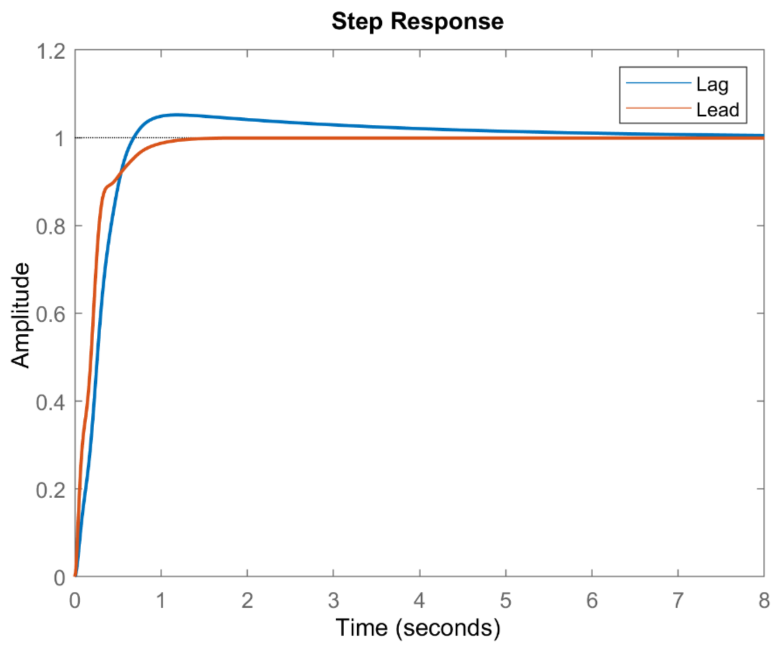
```
Command Window
    SettlingTime: 4.0982
     SettlingMin: 0.9042
     SettlingMax: 1.0520
       Overshoot: 5.1979
      Undershoot: 0
            Peak: 1.0520
        PeakTime: 1.1860
```

همینطور در ادامه نتیجه شبیه‌سازی عملی را داریم:



22) نتیجه شبیه‌سازی عملی سیستم جبران شده با کنترلر Lag

اگر بخواهیم دو کنترلر Lead و Lag طراحی شده را با هم مقایسه کنیم، با مقایسه پاسخ پله آن دو چنانچه در شکل زیر مشاهده می‌شود، واضح است که سرعت کنترلر Lead بیشتر است.



23) مقایسه سرعت دو کنترلر Lead و Lag طراحی شده