گزارش آزمایشگاه کنترل خطی (توپ و میله)

نیما حاجی حیدری (۹۹۲۳۰۱۷)، مهدی شاهینی (۹۹۲۳۰۴۰)

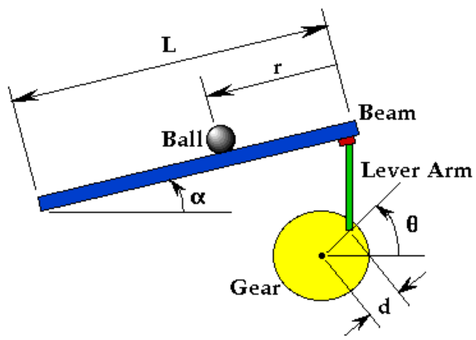
دانشگاه صنعتی امیرکبیر، دانشکده مهندسی برق

۱۴۰۱ - ۱۴۰۲ — ۲

فهرست مطالب

مقدمه

سیستم توپ و میله، تشکیل شده‌است از یک سروو موتور که به میله‌ای با یک انتهای ثابت متصل بوده و از این طریق می‌تواند با تنظیم زاویه‌ی میله با افق، توپ را در نقطه‌ی مشخصی از میله نگه دارد. واضح است که این سیستم، یک سیستم ناپایدار بوده و توپ می‌تواند به راحتی به سمت هر انتهای میله منحرف شده و از کنترل خارج گردد بنابراین از یک سیستم کنترلی مبتنی بر فیدبک بهره برده می‌شود تا موقعیت توپ توسط آن تنظیم گردد.



سه مشخصه مهم در سیستم فوق الذکر وجود دارد :

۱- میزان چرخش زاویه ای سروموتور : θ

۲- زاویه ی میله نسبت به راستای افقی : α

۳- جابجایی خطی توپ در راستای میله : x

با تحلیل دینامیکی سیستم میتوان به تابع تبدیل آن دست‌یافت. در حوزه لاپلاس خواهیم داشت:

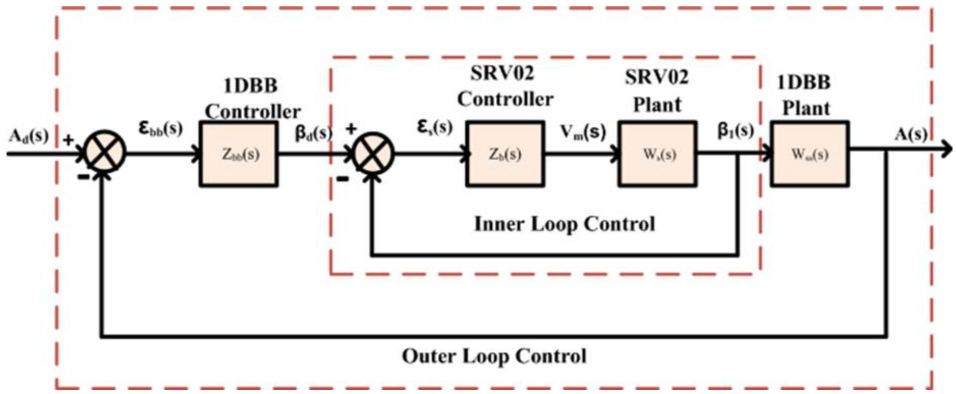
$$G_{1b\&b}\,(s) = \frac{\Theta(s)}{\mathrm{A(s)}} = \frac{L}{\mathrm{r}}$$

$$G_{2b\&b}\,(s) = \frac{X(s)}{A(s)} = \frac{K'_{b\&b}}{s^2}$$

$$K'_{b\&b} = \frac{mgR^2}{mR^2 + J_b}$$

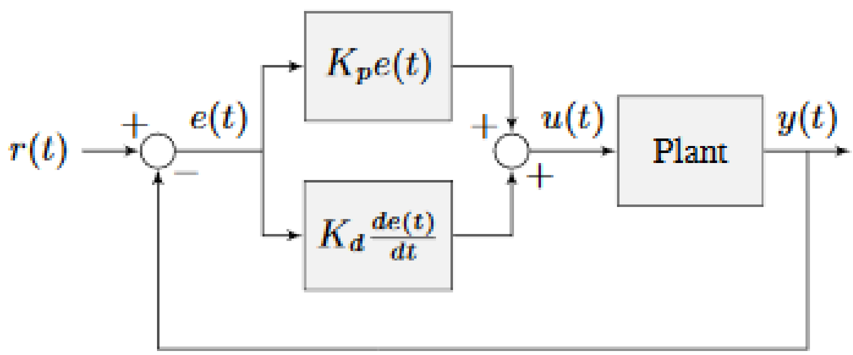
$$G_{b\&b} = \frac{X(s)}{\Theta(s)} = \frac{1}{G_{1b\&b}\,(s)}\,.\,G_{2b\&b}\,(s) = \frac{r}{\mathrm{L}}\,.\frac{K'_{b\&b}}{s^2}$$

همانطور که مشخص است، سیستم گوی و میله به نوعی با سیستم سروو موتور که پیشتر در مورد آن بحث شد ادغام گشته است. به همین علت میتوان بلوک دیاگرام زیر را برای آن ترسیم نمود:

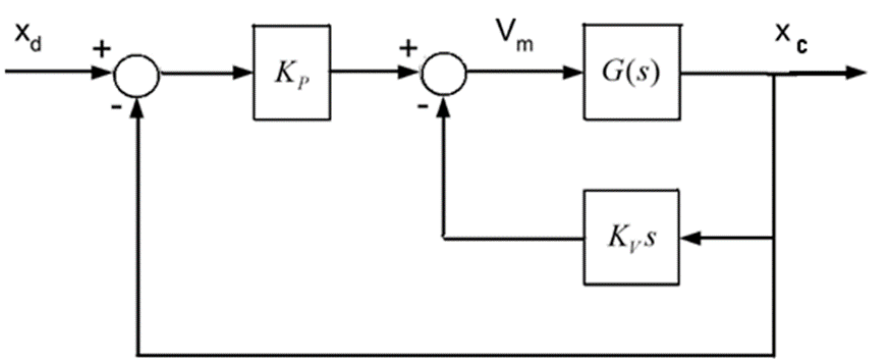


اما کنترل‌کننده‌ی توپ‌ومیله خود می‌تواند به صورت‌های مختلفی باشد که دو حالت آن را بررسی می‌کنیم:

۱- کنترل‌کننده‌های PD :



۲- کنترل‌کننده‌های PV :



از قبل با تابع تبدیل کنترل‌کننده‌های PD آشنا هستیم. در کنترل‌کننده PV نیز خواهیم داشت:

$$\frac{Y(s)}{R(s)} = \frac{K_P G(s)}{1 + (K_P + K_V s)\, G(s)}$$

بدیهی است که می‌توان با در نظر داشتن مشخصات مطلوب، ضرایب مربوط به کنترل‌کننده‌ها را تعیین کرد.

طراحی کنترل‌کننده PV در حالت کلی

پیشتر دیدیم که می‌توان یک کنترل‌کننده‌ی PV را به چه صورت نشان داد. حال از همان بلوک دیاگرام خواهیم داشت:

$$U(s) = K_P \,.\, E(s) - K_V s \,.\, Y(s)$$

$$Y(s) = G(s)\,.\, U(s)$$

$$E(s) = R(s) - Y(s)$$

با جایگذاری از روابط بالا خواهیم داشت:

$$Y(s) = G(s)\,.\, K_P \,.\, E(s) - G(s)\,.\; K_V s \,.\, Y(s)\;.$$

$$Y(s) = G(s)\,.\, K_P . R(s) - G(s). K_P \,.\, Y(s) - G(s).\; K_V s \,.\, Y(s)$$

می‌توان تابع تبدیل را به فرم زیر تعریف نمود:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{G(s)\,.\, K_P}{1 + \; G(s)\,.\, K_P + G(s).\; K_V s}$$

در سیستم می‌توان با جایگذاری مقادیر داخلی تابع تبدیل G گفت که $G(s) = \frac{7}{s^2}$. بنابراین داریم:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{\frac{7}{s^2}\,.\, K_P}{1 + \frac{7}{s^2}\,.\, K_P + \frac{7}{s^2}\;\; K_V s}$$

$$T(s) = \frac{Y(s)}{R(s)} = \frac{7\,.\, K_P}{s^2 + 7K_v\, s + 7\;\; k_p}$$

از سوی دیگر ولی می‌دانیم که در سیستم‌های مرتبه دو می‌توان تابع تبدیل را به صورت زیر نمایش داد:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\,\zeta\omega_n\, s + \;\; \omega_n^2}$$

پس با برابر قرار دادن دو رابطه آخر می‌توان به ارتباط بین ضرایب کنترل‌کننده و مشخصات سیستم دست یافت. بنابراین، خواهیم داشت:

$$7K_v = 2\,\zeta\omega_n$$
$$7K_p = \omega_n^2$$

بنابراین پس از ساده شدن، خواهیم داشت :

$$K_P = \frac{\omega_n^2}{7}$$
$$K_V = 2\frac{\zeta\omega_n}{7} = 2 * \zeta * \frac{\sqrt{K_p}}{\sqrt{7}}$$

طراحی کنترل‌کننده بر اساس شاخص‌های عملکردی مطلوب

با توجه به روابط به دست آمده اگر بخواهیم سیستمی طراحی کنیم که اولین پیک آن، پس از ۱.۵ ثانیه رخ داده و بیشترین میزان بالا زدگی آن، ۴.۶ درصد باشد. می‌دانیم که میتوان این پارامترها را به صورت زیر به مشخصات سیستم مربوط کرد:

$$P.O. = 100\, e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}$$
$$T_P = \frac{\pi}{\omega_n \cdot \sqrt{1-\zeta^2}}$$

با حل همزمان دو معادله بالا خواهیم داشت :

$$\omega_n = 2.9619, \quad \zeta = 0.7071$$

به خاطر داریم که ضرایب کنترل‌کننده‌ی PV با ضریب میرایی و فرکانس طبیعی، در ارتباط بودند. با اعمال روابط بدست آمده بین ضرایب و مشخصات سیستم، خواهیم داشت:

$$K_P = 1.2533$$
$$K_V = 0.5984$$

تابع تبدیلی که پیشتر بدست آمد، به فرم زیر خواهد بود:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{8.773}{s^2 + 4.189\,s + 8.773}$$

می‌توان با استفاده از شبیه‌سازی، شاخص‌های عملکردی را بدست آورده و پاسخ سیستم به ورودی‌های مختلف را روی نمودار نمایش داد.

شبیه سازی کامپیوتری

بـرای مشـاهده پاسـخ سیسـتم جبران‌شـده بـا بهره‌هـای مـورد محاسـبه در بخـش قبـل، می‌تـوان از رشته کدهای زیر استفاده کرد.

Desirable Conditions

$$\varsigma = \frac{1}{\sqrt{2}}, \mathrm{T}_p = 1.5\,\mathrm{sec}, \omega_n = \frac{\pi}{\mathrm{T}_p\sqrt{1-\zeta^2}} = \frac{\pi\sqrt{2}}{\mathrm{T}_p}$$

```
zeta = sqrt(2) / 2
```

```
zeta = 0.7071
```

```
t_p = 1.5
```

```
t_p = 1.5000
```

```
w_n = sqrt(2) * pi / t_p
```

```
w_n = 2.9619
```

$\mathrm{K}_p, \mathrm{K}_v$

```
k_p = (w_n^2) / 7
```

```
k_p = 1.2533
```

```
k_v = 2 * zeta * sqrt(7 * k_p) / 7
```

```
k_v = 0.5984
```

Building Transfer Function

```
T = tf(7*k_p, [1 7*k_v 7*k_p])
```

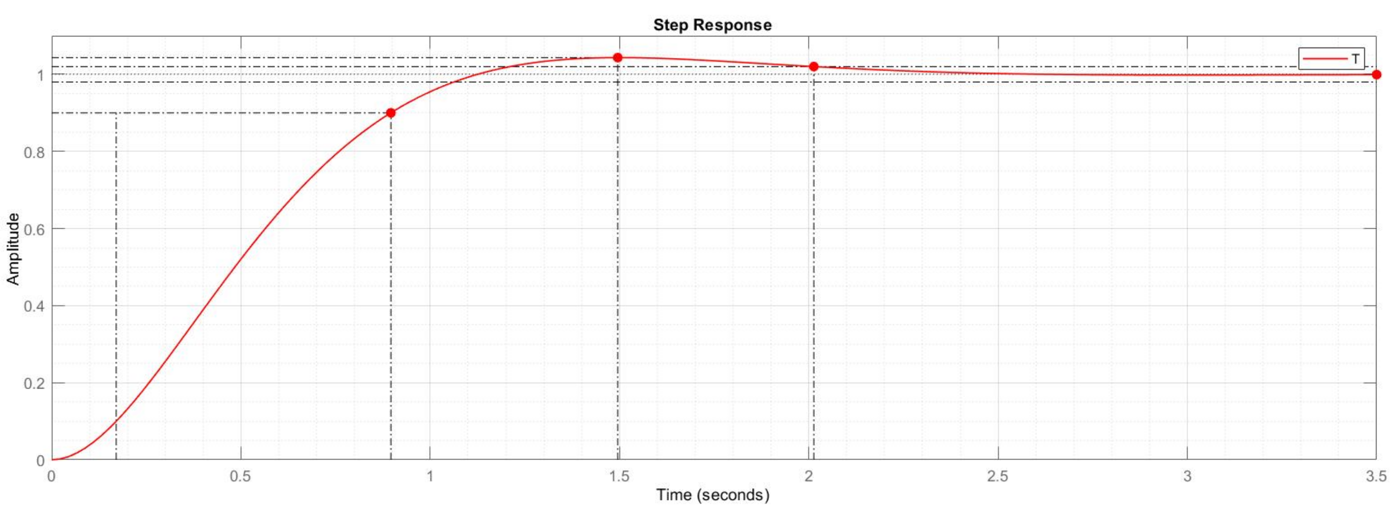
```
T =

          8.773
  ---------------------
  s^2 + 4.189 s + 8.773

Continuous-time transfer function.
```

بنابراین در پایان این مرحله، تابع تبدیل سیستم جبران‌شده در دسترس است. پس از این، می‌توان با استفاده از تابع stepinfo(T) مشخصات تابع تبدیل را مشاهده کرده و با استفاده از step(T) نمودار پاسخ زمانی سیستم به ورودی پله را رسم نمود.

نتایج شبیه سازی پاسخ پله بصورت زیر است؛ با توجه به نتایج، درستی طراحی مشخص می‌گردد چرا که درصد بالا زدگی و همچنین زمان رسیدن پاسخ به مقدار بیشینه برای اولین بار، بسیار نزدیک به مشخصات مطلوب مسئله می‌باشد.

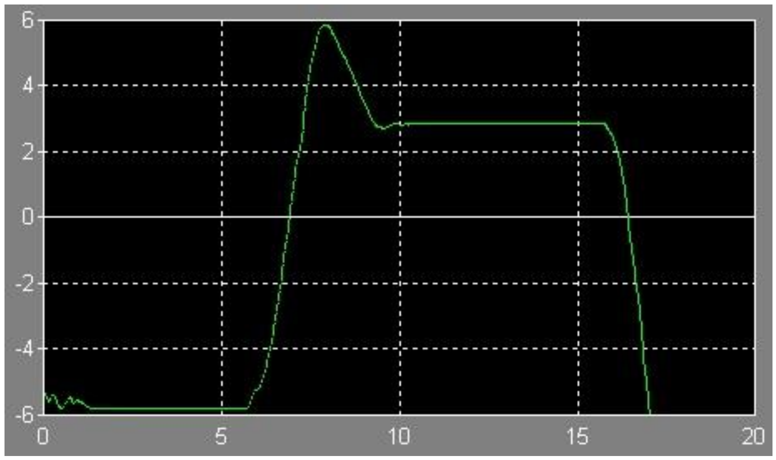
```
stepinfo(T)
```

```
ans = struct with fields:
         RiseTime: 0.7255
     SettlingTime: 2.0131
      SettlingMin: 0.9034
      SettlingMax: 1.0432
        Overshoot: 4.3210
       Undershoot: 0
             Peak: 1.0432
         PeakTime: 1.4952
```



نتیجه عملی

پس از اعمال کنترل‌کننده به سیستم واقعی، نتایج زیر بدست آمده است:



بحث و نتیجه گیری

۱- بعد از طراحی کنترل‌کننده و مشاهده پاسخ، آیا مقدار overshoot و time rise سیگنال‌های simulated ball position و Measured ball position با هم و با مقدار مطلوب تفاوت داشت؟ اگر جواب مثبت است دلیل آن را توضیح دهید. به نظر شما آیا راه حلی وجود دارد که این اختلاف به حداقل برسد؟ توضیح دهید.

مطابق با سوالی که در دستور کار پرسش‌شده، مشاهده می‌شود که تفاوت بسیار زیادی بین نتایج عملی و نتایج حاصل از شبیه سازی وجود دارد. علت‌های متفاوتی را می‌توان برای توجیه این اختلاف، بیان کرد:

الف) شرایط در هنگام طراحی به صورت ایده‌آل در نظر گرفته می‌شود.

ب) دقیق نبودن دستگاه‌های اندازه‌گیری می‌تواند بر پاسخ تاثیر بگذارد.

ج) خطی‌سازی سیستم، می‌توانسته است حول نقطه تعادل دقیق‌تری صورت گیرد.

۲- از چه کنترل‌کننـده‌های دیگـری می‌توانسـتید اسـتفاده کنیـد؟ دلیـل اسـتفاده از کنترل‌کننـده PV را توضیح دهید.

در پاسخ باید گفت که برای این سیستم می‌توان از کنترل‌کننده‌های Lead-Lag هم استفاده کرد ولی به این علت از کنترل‌کننده PV استفاده می‌کنیم که وجود مشتق گیر در مسیر پیشرو، از تغییرات ناگهانی مشتق گرفته و باعث می‌شود که در سیستم، یک رفتار بسیار نزدیک به ضربه ظاهر گردد. این رفتار، می‌تواند به سیستم و همچنین تجهیزات اندازه‌گیری آسیب برساند. پس بنابراین بلوک مشتق‌گیر به فیدبک منتقل شده و تشکیل یک کنترل‌کننده PV می‌دهند که مشکل بیان‌شده را رفع می‌کند. دلیل دیگر استفاده از کنترل‌کننده PV ، شبیه بودن تابع تبدیل سیستم به تابع تبدیل استاندارد یک سیستم درجه دو نوعی می‌باشد.

لازم به ذکر است با توجه به اینکه تابع تبدیل کلی پلنت دو قطب مضاعف در ۰ دارد به تبعات آن مشکلات این طراحی نیز پیش می‌آید.

همچنین اگر از ناهنجاری‌های گفته شده صرف نظر شود و از کنترل کننده lead استفاده کنیم و صفر آن را بسیار نزدیک به مبدا یا روی مبدا قرار دهیم نمیتوان شرایط مطلوب را به صورت دقیق ارضا کنیم.

علت آن این است که :

$$C(s) = \frac{s}{s+p}, G = \frac{7}{s^2}, \qquad T = \frac{7}{s^2+ps+7}$$

$$\omega_n = \sqrt{7}$$

از طرفی طبق فرض خواهیم داشت:

$$\zeta = \frac{1}{\sqrt{2}}, \quad T_P = \frac{\pi}{\omega_n \,.\sqrt{1-\zeta^2}} \rightarrow T_P = \frac{\pi}{\sqrt{7}\,.\sqrt{1-\zeta^2}} = 2.1381$$

از طرفی طبق فرض داریم: $T_p = 1.5\ sec$

که در آن صورت $\zeta = 0.6110$ خواهد بود.

بنابراین مجبور به انجام یک تریدآف بین پاسخ ماندگار و بالا زدگی خواهیم بود!

نتایج این کنترل‌کننـده در صـفحه بعـد آمـده اسـت! (نامناسـب بـودن ایـن کنترل‌کننـده بـه وضـوح پیداست!)

Ball&Beam with Lead Controller

Desirable Conditions

$\omega_n = \sqrt{7}, \mathrm{T}_p = 1.57\,\mathrm{sec}, 1 - \dfrac{\pi}{\mathrm{T}_p\,\omega_n} = \zeta^2 = 1 - \dfrac{\pi}{\mathrm{T}_p\sqrt{7}}$

```
t_p = 2.2
```

```
t_p = 2.2000
```

```
w_n = sqrt(7)
```

```
w_n = 2.6458
```

```
zeta = sqrt(1-(pi/(t_p*w_n)))
```

```
zeta = 0.6784
```

```
p = 2 * zeta * w_n
```

```
p = 3.5899
```

```
s = tf('s');
```

```
C = s / (s+p)
```

```
C =

      s
  --------
  s + 3.59

Continuous-time transfer function.
```

```
G = 7 / s^2
```

```
G =

   7
  ---
  s^2

Continuous-time transfer function.
```

```
T_cl = (C*G)/(1 + C*G)
```

```
T_cl =

          7 s^4 + 25.13 s^3
  --------------------------------------
  s^6 + 7.18 s^5 + 19.89 s^4 + 25.13 s^3

Continuous-time transfer function.
```

```
T = 7 / (s^2 + s*p + 7)
```

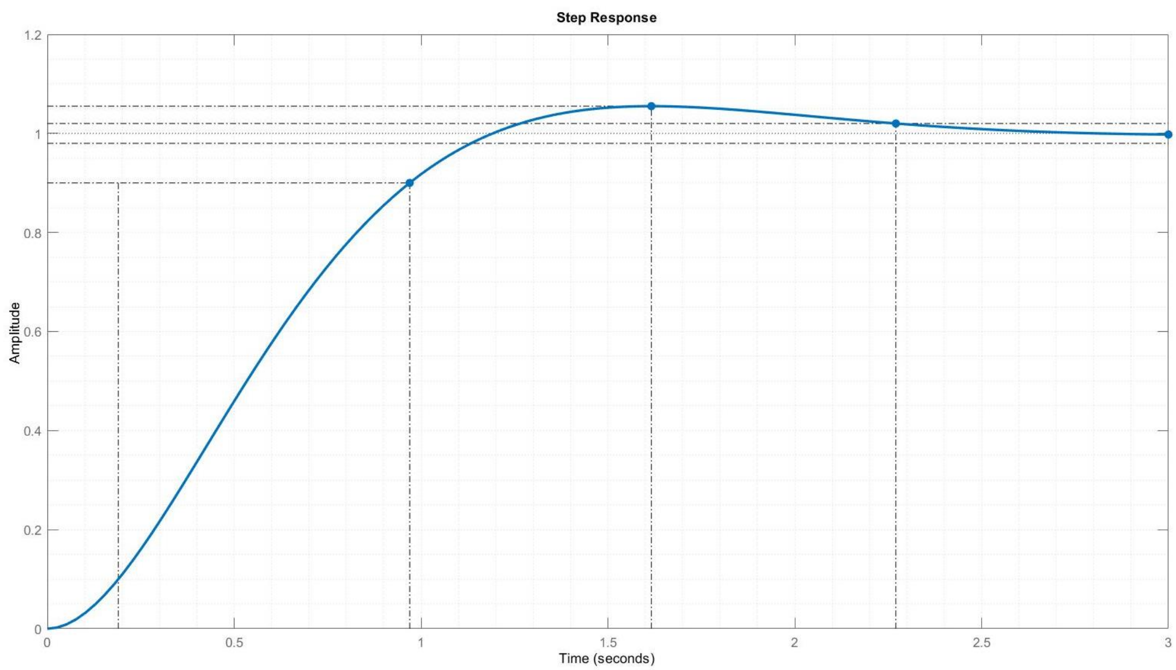
```
T =

        7
  ----------------
  s^2 + 3.59 s + 7

Continuous-time transfer function.
```

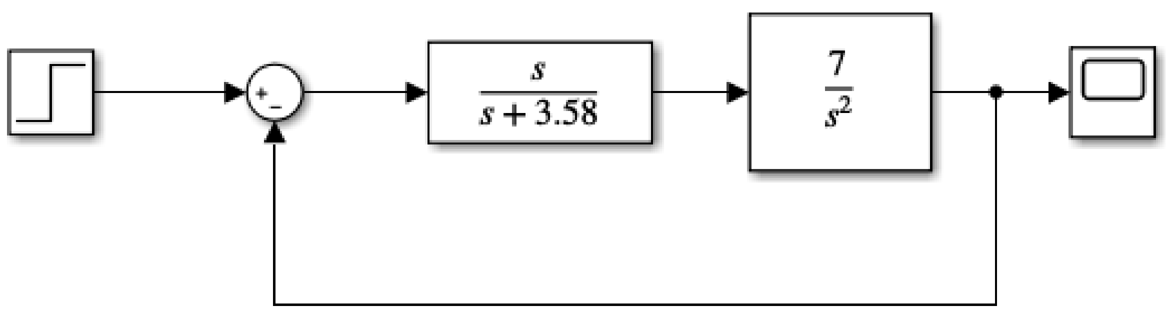
```
T_cl/T
```

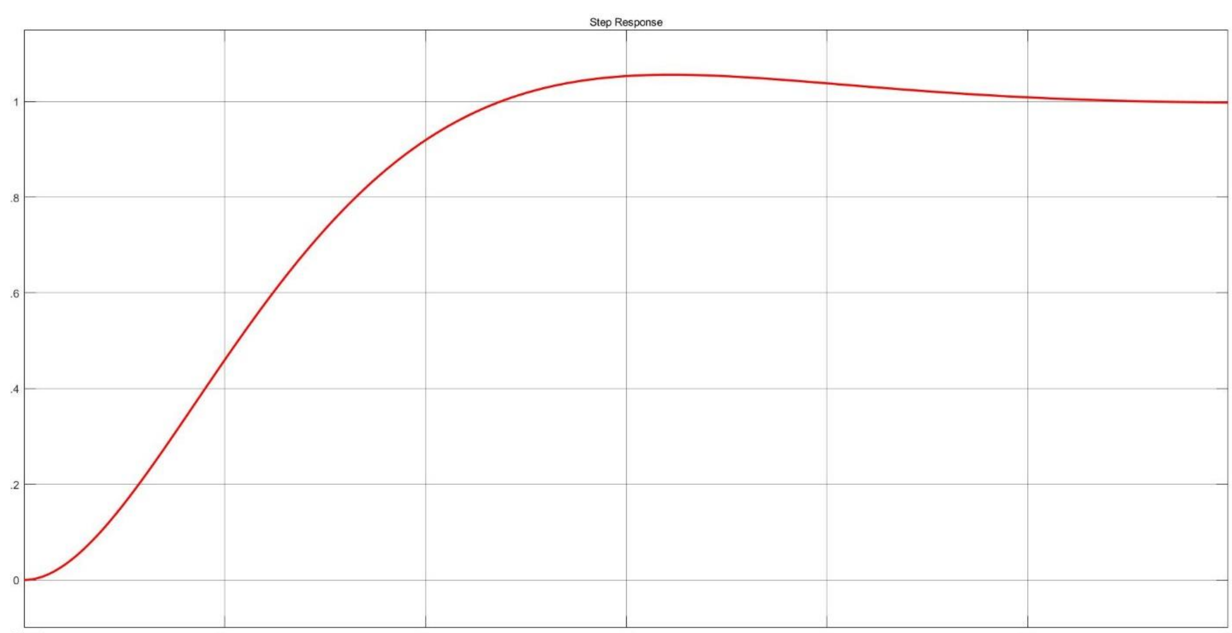
```
ans =

  7 s^6 + 50.26 s^5 + 139.2 s^4 + 175.9 s^3
  -----------------------------------------
  7 s^6 + 50.26 s^5 + 139.2 s^4 + 175.9 s^3

Continuous-time transfer function.
```

```
stepinfo(T)
```

```
ans = struct with fields:
        RiseTime: 0.7798
    SettlingTime: 2.2705
     SettlingMin: 0.9032
     SettlingMax: 1.0550
       Overshoot: 5.4961
      Undershoot: 0
            Peak: 1.0550
        PeakTime: 1.6163
```


Step Response
Amplitude
Time (seconds)
1.2
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
2.5
3


+
–
$\frac{s}{s+3.58}$
$\frac{7}{s^2}$


Step Response
1
.8
.6
.4
.2
0
Offset=0
0.5
1
1.5
2
2.5
3

۳- آیا میتوان به جای استفاده از کنترل‌کننده Lead برای کنترل موقعیت موتور نیز از کنترل‌کننده‌ی PV استفاده نمود؟ اگر بله به لحاظ تئوری طراحی و شبیه‌سازی نمایید و همچنین ملاحظات عملی را نیز بررسی نمایید. اما اگر پاسخ منفی است، چرا؟

کنترل‌کننده PV قابلیت طراحی با مکان هندسی برای پلنت سروو موتور ندارد!

همچنین حلقه داخلی بسیار سریع بوده و پاسخ سریعا در مقدار مطلوب میرا میشود. این امر سبب میشود مشتق گرفتن از پاسخ در لحظه اول به ورودی، سیگنالی شبیه به ضربه اضافه کند و در لحظات بعد چون پاسخ به مقدار مطلوب رسیده است، تغییرات سیگنال خروجی صفر و فیدبک قطع شود! این مسائل ما را به پرهیز از کنترل‌کننده PV سوق می‌دهد.

اگر از لحاظ تئوری ریاضیات قصد حل کردن داشته‌باشیم خواهیم داشت:

$$T_{cl} = \frac{K_p G_{servo}(s)}{1 + \left(K_p + K_v s\right) G_{servo}(s)}$$

$$\xRightarrow[G_{srv}(s)=\frac{64.2}{s^2+36.42s}]{} \frac{K_p * 64.2}{s^2 + (36.42 + 64.2K_v)s + K_p * 64.2} \rightarrow$$

$$\omega_n = \sqrt{K_p * 64.2}$$

$$2 * \zeta\omega = 36.42 + 64.2K_v$$

$$Desired: T_s = \frac{4}{\zeta\omega} = 0.1, \quad P.O < 4.7\% \rightarrow \zeta > \frac{1}{\sqrt{2}}$$

$$\omega = \frac{40}{\frac{1}{\sqrt{2}}} = 40\sqrt{2} \rightarrow K_p = 49.8442$$

$$2 * 40 = 36.42 + 64.2K_v \rightarrow K_v = 0.6788$$

Servi Motor with PV

Desired

```
t_s = 0.1
```

```
t_s = 0.1000
```

```
zeta = 1 / sqrt(2)
```

```
zeta = 0.7071
```

```
w_n = 4 / (t_s * zeta)
```

```
w_n = 56.5685
```

```
K_p = w_n^2 / 64.2
```

```
K_p = 49.8442
```

```
K_v = ((2 * (4 / t_s)) - 36.42) / 64.2
```

```
K_v = 0.6788
```

G

servo motor

```
G = tf(64.2, [1 36.42 0])
```

```
G =

      64.2
  -------------
  s^2 + 36.42 s

Continuous-time transfer function.
```

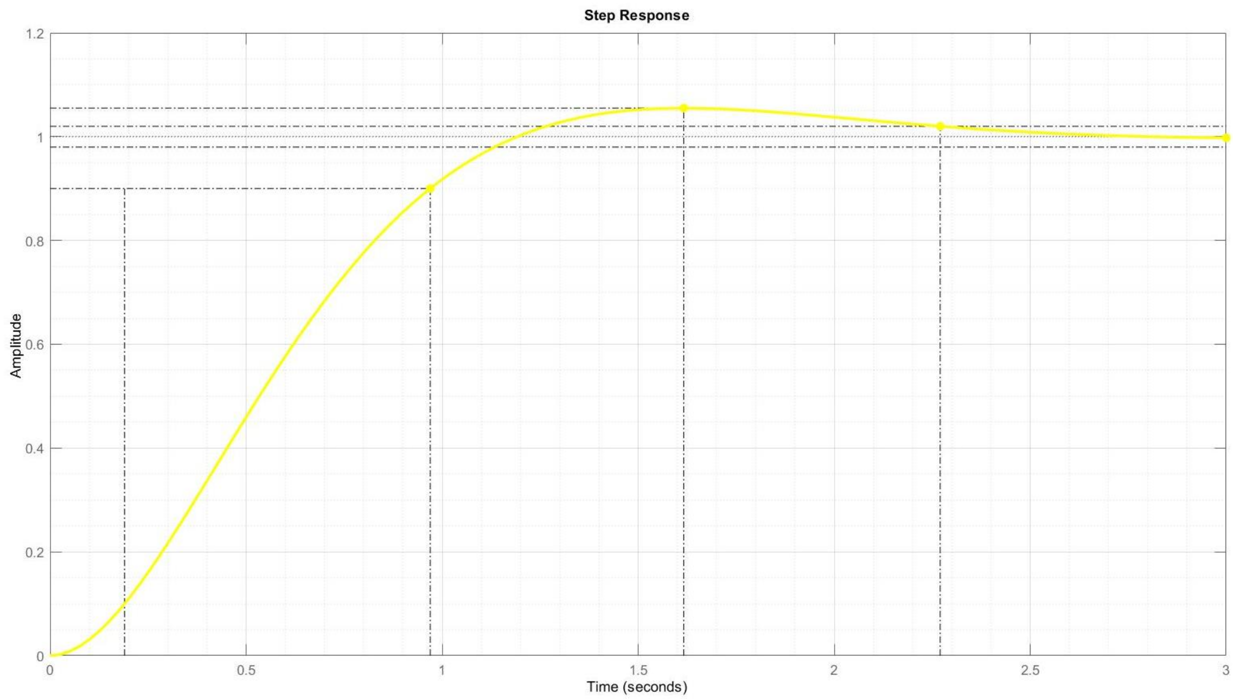
Transfer Function

```
T_cl  = K_p * G / (1 + (K_p + K_v * s) * G)
```

```
T_cl =

             3200 s^2 + 1.165e05 s
  ---------------------------------------
  s^4 + 116.4 s^3 + 6114 s^2 + 1.165e05 s

Continuous-time transfer function.
```

```
stepinfo(T)
```

```
ans = struct with fields:
        RiseTime: 0.7798
    SettlingTime: 2.2705
     SettlingMin: 0.9032
     SettlingMax: 1.0550
       Overshoot: 5.4961
      Undershoot: 0
            Peak: 1.0550
        PeakTime: 1.6163
```



به راحتی قابل مشاهده است که به جای پاسخ سریع و بالازدگی کم در حالت lead به زمان نشست طولانی‌تر و بالا زدگی بیشتر دچار شد!

۴- آیا با عدم صرف نظر از حلقه‌ی داخلی همچنان می‌توان از PV برای حلقه‌ی خارجی استفاده نمود؟ اگر بله به لحاظ تئوری طراحی و شبیه‌سازی نمایید. اما اگر پاسخ منفی است یک کنترل‌کننده برای این منظور پیشنهاد داده و عملکرد آن را بررسی کنید.

بله. میتوان استفاده نمود. لازم به ذکر است اگر از حلقه داخلی صرفه‌نظر نکنیم شاهد تابع تبدیلی پیچیده خواهیم بود. این تابع تبدیل پیچیده با استفاده از نرم‌افزار متلب یا تقریب‌های موجود در کنترل خطی باز هم به تابع تبدیل مراحل قبل تبدیل می‌شود و کنترل کننده PV به صورت تئوری پاسخگو خواهد بود.

همچنین در صورتی که در G_{plant} اگر تابع تبدیل غیر تقریبی را قرار دهیم و بر اساس همان ضرایب عادی کنترل‌کننده PV تابع تبدیل را بدست آورده و ویژگی‌هایش را مشاهده کنیم همان خواسته‌های مطلوب بدست خواهد آمد!

مراحل:

۱- طراحی کنترل‌کننده lead برای سروو موتور

۲- در نظر گرفتن حلقه سروو موتور

۳- نوشتن تابع تبدیل جدید کل سیستم

۴- کد نویسی تمامی راه‌حل و شبیه‌سازی!

$$T_s = 0.1 = \frac{4}{\zeta\omega_n} \rightarrow \zeta\omega_n = 40$$

$$P.O < 4.7\% \rightarrow \zeta > \frac{1}{\sqrt{2}} \rightarrow 40\sqrt{2} > \omega_n$$

$$s_d = -\zeta\omega_n \pm \omega_n\sqrt{1-\zeta} = -40 \pm j40$$

We set the zero of the controller in the true value of the desired poles and obtain the angle condition for the controller pole:

$$90° - \left(\operatorname{atan}\frac{40}{p-40} + 90° + \operatorname{atan}\frac{40-36.42}{40} + 90° + \operatorname{atan}\frac{40}{40}\right) = 180°$$

$$p = 87.8638$$

If we consider the gain of the controller as 57

$$C_{lead} = 57 * \frac{s+40}{s+87.87}$$

$$T_{cl_{srv}} = \frac{C_{lead}G_{srv}}{1 + C_{lead}G_{srv}}$$

$$= \frac{57 * (s + 40) * (64.2)}{\big((s + 87.87) * (s^2 + 36.42s)\big) + \big(57 * (s + 40) * (64.2)\big)}$$

$$= \frac{3659s + 146376}{s^3 + 123.9s^2 + 3185s}$$

$$G_{plant} = T_{cl_{srv}} * G_{bb} = \frac{3659s + 146376}{s^3 + 123.9s^2 + 3185s} * \frac{7}{s^2}$$

$$T(s) = \frac{Y(s)}{R(s)} = \frac{G(s)\,.K_P}{1 + \ G(s)\,.K_P + G(s).\ K_V s}$$

$$= \frac{7 * (3659s + 146376)}{s^2 * (s^3 + 123.9s^2 + 3185s) + 7 * (3659s + 146376)}$$

$$= \frac{7 * (s + 40)}{\frac{s^3}{3659} * (s + 35.877)(s + 88.023) + 7 * (s + 40)}$$

$$\approx \frac{7}{\frac{s^3}{3659}(s + 88.023) + 7}$$

Ball&Beam

Considering the servo motor

```
close all
clear
clc
s = tf('s');
```

Desired servo motor conditions

$\zeta = \frac{1}{\sqrt{2}}, \mathrm{T}_s = 0.1 \text{ sec}, s_d = -\zeta\omega_n \pm \omega_n\sqrt{1-\zeta^2}$

```
t_s = 0.1
```

```
t_s = 0.1000
```

```
zeta = 1 / sqrt(2)
```

```
zeta = 0.7071
```

```
w_n = 4 / (t_s * zeta)
```

```
w_n = 56.5685
```

```
s_d1 = (-4/t_s) + (1j * w_n * sqrt(1-zeta^2))
```

```
s_d1 = -40.0000 + 40.0000i
```

```
s_d2 = s_d1'
```

```
s_d2 = -40.0000 - 40.0000i
```

G servo motor

```
G_srv = 64.2 / (s * (s + 36.42))
```

```
G_srv =

      64.2
  -------------
  s^2 + 36.42 s

Continuous-time transfer function.
```

Lead Controller

```
k_lead = 57;
z_lead = 40;
p_lead = 87.8638;

C_lead = k_lead * (s+z_lead)/(s+p_lead)
```

```
C_lead =

  57 s + 2280
  -----------
   s + 87.86

Continuous-time transfer function.
```

Servo Motor Loop

```
T_srv = feedback(series(C_lead, G_srv), 1)
```

```
T_srv =

             3659 s + 146376
  ---------------------------------
  s^3 + 124.3 s^2 + 6859 s + 146376

Continuous-time transfer function.
```

G ball&beam

```
G_bb = 7 / s^2
```

```
G_bb =

   7
  ---
  s^2

Continuous-time transfer function.
```

PV Controller

from BB_livecode.mlx

```
K_p = 1.2533
```

```
K_p = 1.2533
```

```
K_v = 0.5984
```

```
K_v = 0.5984
```

System

```
G_plant = series(T_srv, G_bb);
L_in = feedback(G_plant, K_v*s);
T_ol = series(K_p, L_in);
T_cl = feedback(T_ol, 1)
```

```
T_cl =

                    3.21e04 s + 1.284e06
  -----------------------------------------------------------------
  s^5 + 124.3 s^4 + 6859 s^3 + 1.617e05 s^2 + 6.452e05 s + 1.284e06

Continuous-time transfer function.
```

STEP

```
stepinfo(T_cl)
```

```
ans = struct with fields:
        RiseTime: 0.6903
    SettlingTime: 1.9393
     SettlingMin: 0.9021
     SettlingMax: 1.0431
       Overshoot: 4.3141
      Undershoot: 0
            Peak: 1.0431
        PeakTime: 1.4441
```

به وضوح هویداست کـه تمـامی مـوارد مطلـوب کـاملا ارضـا شـده و در نظـر گـرفتن یـا نگـرفتن سروو موتور تاثیری در پاسخ نهایی نداشته است!

۵- بر اساس پاسخی که به سوال قبلی میدهید با قراردادن منحنیهای خروجی بر روی یکدیگر، اثر ایدهآل در نظر گرفتن حلقه داخلی را بررسی کنید.