به نام خدا

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

عنوان:

Differential Drive Mobile Robot

نام درس:

کنترل مدرن

نام استاد:

دکتر هاجر عطریانفر

مهندس الناز فیروزمند

نگارش :

محمد برابادی

مارال مرداد

محیا حقگو

سجاد قدیری

فهرست صفحه

فاز اول

معرفی

در سال های اخیر مطالعات و استفاده از ربات های متحرک بسیار مد نظر گرفته است. هر ساله با پیشرفت روش های کنترلی در زمینه های تئوری و عملی،کاربرد این ربات ها بیشتر نمایان می‌شود.

از کاربرد های این ربات می‌توان استفاده در مسیریابی در بیمارستان‌ها، فروشگاه‌های مواد غذایی، نظافت خانگی و ... نام برد. ربات مورد استفاده ما یک ربات ۴ چرخ غیرهولونومیک می‌باشد.

بیان مفهوم غیرهولونومیک بسیار مفصل و نیازمند دانشی درباره درجه آزادی (degree of freedom) و فضای پیکربندی (configuration space) است.از رابطه گرابلر برای درجه آزادی داریم:

$$dof = \sum (freedom\ of\ bodies) - \#\ of\ independent\ constraints$$

$$dof = m(N - ۱ - J) + \sum_{i=۱}^{J} f_i$$

که در آن N تعداد بدنه‌های شامل زمین، J تعداد مفاصل، m تعداد درجه آزادی تک بدنه و f_i ها درجه آزادی مفصل ها می‌باشد. توصیف سیستم هایی که درجه آزادی پایینی دارند (مثلا یک) و تعداد اجزای آنها زیاد می‌باشد بسیار سخت است.

در این مواقع سیستم را به درجات بالاتر اقلیدسی (higher dimensional Euclidean) می‌برند و با قرار دادن قیدهای حلقه بسته سیستم، آن را راحت تر توصیف می‌کنند. مثلا برای یک سیستم با درجه آزادی یک که دارای ۴ لینک حرکتی می‌باشد می‌توان با افزایش بعد آن به ۴ و قرار دادن ۳ قید سیستم را بهتر توصیف کرد.

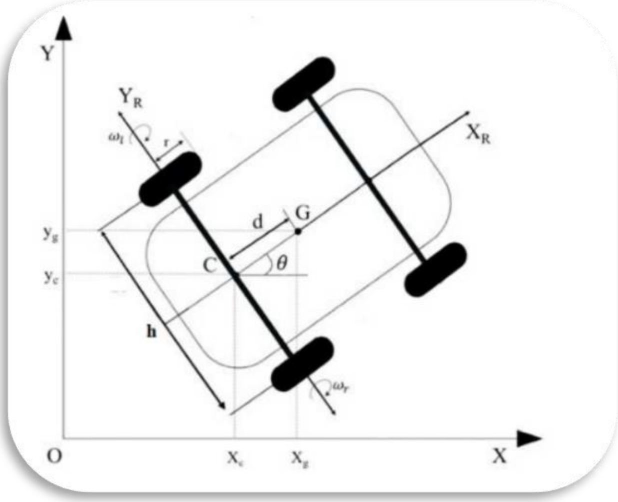
اگر این قید ها بعد فضای پیکربندی را کاهش دهند به آنها holonomic و درغیر این صورت به آنها nonholonomic می‌گویند.

از لحاظ فیزیکی بیانگر آن هست که ربات نمی‌تواند به طور مستقیم در جهت نشان داده شده در شکل حرکت کند . برای تحقق حرکت در این جهت باید از parallel parking استفاده نماید .

ربات از دو جفت چرخ متصل به موتور استفاده می کند. هرجفت که با یک خط به هم متصل هستند، در جهت یکسان دوران می‌کنند. نکته ی مهم در این ربات ها این است که چرخ ها در هرجفت باید با سرعت یکسان بچرخند ، در غیر این صورت آن چرخی که سرعتش کمتر است، هرز و ناکارآمد خواهد بود و ربات در مسیر مستقیم حرکت نخواهد کرد.

هدف این پروژه بررسی ساختار دینامیکی و سینماتیکی ربات مذکور و طراحی کنترلر مناسبی برای حرکت ربات به منظور تعقیب یک مسیر دایره‌ای بوده است.

بررسی مدل غیرخطی



برای بررسی معادلات غیرخطی از معادلات سینماتیکی استفاده شده است. در ادامه به شرح نحوه به دست آمدن معادلات غیرخطی سینماتیکی سیستم می پردازیم. همانطور که در شکل قابل مشاهده است، برای حرکت ربات دو محور مختصات قابل تعریف است که شامل محور مختصات سراسری (x , y) و محور مختصات محلی ربات می‌باشد. این دو محور مختصات از طریق یک ماتریس تبدیل دوران به یکدیگر مرتبط می‌شوند.

در شکل بالا X , Y محور مختصات سراسری و Y_R , X_R محور مختصات محلی ربات است.

$$\dot{x} = \frac{r(\omega_r+\omega_l)}{۲}\cos\theta \qquad (۱)$$

$$\dot{y} = \frac{r(\omega_r+\omega_l)}{۲}\sin\theta \qquad (۲)$$

$$\dot{\theta} = \frac{r(\omega_r-\omega_l)}{h} \qquad (۳)$$

w_r و w_l سرعت های زاویه ای چپ و راست چرخ‌ها هستند. h فاصله ی مراکز چرخ ها از یکدیگر است. θ زاویه ی مرکز ربات با محور X محور مختصات سراسری می‌باشد.

برای به دست آوردن معادلات حالت سیستم ورودی های کنترلی سرعت خطی چرخ های چپ و راست در نظر گرفته شده اند.

$$input = \begin{bmatrix} r\omega_l \\ r\omega_r \end{bmatrix}$$

و همین طور لغزش به صورت زیر مدل می‌شود:

$$i = ۱ - \frac{v}{r\omega}$$

که r و v به ترتیب شعاع چرخ ها و سرعت خطی ربات می‌باشند.

اگر بخواهیم معادلات ۱ و ۲ و ۳ را به صورت ماتریسی بنویسیم، ماتریس زیر به دست می‌آید:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & ۰ \\ \sin\theta & ۰ \\ ۰ & ۱ \end{bmatrix} \begin{bmatrix} \frac{۱-i_l}{۲} & \frac{۱-i_r}{۲} \\ -\frac{۱-i_l}{h} & \frac{۱-i_r}{h} \end{bmatrix} \begin{bmatrix} r\omega_l \\ r\omega_r \end{bmatrix}$$

برای معادلات بالا ترم خطا تشکیل می‌دهیم. ماتریس خطا به صورت زیر است:

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & ۰ \\ -\sin\theta & \cos\theta & ۰ \\ ۰ & ۰ & ۱ \end{bmatrix} \begin{bmatrix} x^{ref} - x \\ y^{ref} - y \\ \theta^{ref} - \theta \end{bmatrix}$$

برای به دست آوردن معادلات حالت باید از معادلات خطا مشتق گرفته شود.

برای ترم e_x به صورت زیر می‌نویسیم :

$$e_x = \cos\theta\left(x^{ref} - x\right) + \sin\theta\left(y^{ref} - y\right)$$

$$\dot{e}_x = -\dot{\theta}\sin\theta\left(x^{ref} - x\right) + \dot{\theta}\cos\theta\left(y^{ref} - y\right) + \cos\theta\left(\dot{x}^{ref} - \dot{x}\right) + \sin\theta\left(\dot{y}^{ref} - \dot{y}\right)$$

با در نظر گرفتن

$$\dot{x}^{ref} = v^{ref}\cos\theta^{ref}$$

$$\dot{x} = v\cos\theta$$

$$\dot{y}^{ref} = v^{ref}\sin\theta^{ref}$$

$$y = v\sin\theta$$

$$e_y = -\sin\theta\left(x^{ref} - x\right) + \cos\theta\left(y^{ref} - y\right)$$ و

خواهیم داشت :

$$\dot{e_x} = \dot{\theta}e_y + v^{ref}\cos\left(\theta^{ref} - \theta\right) - v$$

برای ترم e_y به صورت زیر می‌نویسیم :

$$e_y = -\sin\theta\left(x^{ref} - x\right) + \cos\theta\left(y^{ref} - y\right)$$

$$\dot{e_y} = -\dot{\theta}\cos\theta\left(x^{ref} - x\right) - \dot{\theta}\sin\theta\left(y^{ref} - y\right) - \sin\theta\left(\dot{x}^{ref} - \dot{x}\right) + \cos\theta\left(\dot{y}^{ref} - \dot{y}\right)$$

با در نظر گرفتن

$\dot{x}^{ref} = v^{ref} \cos\theta^{ref}$

$\dot{x} = v\cos\theta$

$\sin\theta \;\; \dot{y}^{ref} = v^{ref}$

$y = v\sin\theta$

$e_x = \cos\theta\left(x^{ref} - x\right) + \sin\theta\left(y^{ref} - y\right)$

خواهیم داشت:

$$\dot{e_y} = -\dot{\theta}e_x + v^{ref}\sin\left(\theta^{ref} - \theta\right)$$

برای ترم e_θ به صورت زیر می‌نویسیم:

$$\dot{e_\theta} = \dot{\theta}^{ref} - \dot{\theta}$$

خطی‌سازی سیستم سینماتیکی

برای خطی سازی معادلات غیر خطی ارائه شده نقطه تعادل $\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} ۰ \\ ۰ \\ ۰ \end{bmatrix}$ در نظر گرفته شده است. در اصل خطا نزدیک صفر در نظر گرفته می‌شود اگر دقیقا صفر در نظر گرفته شود دیگر نمی‌توان کنترلر طراحی کرد.

با جایگذاری نقطه تعادل در معادلات قسمت قبل ، معادلات حالت سیستم بر اساس ترم های خطا به صورت زیر درمی‌آید:

$$\begin{bmatrix} \dot{e_x} \\ \dot{e_y} \\ \dot{e_\theta} \end{bmatrix} = \begin{bmatrix} ۰ & \dot{\theta}^{ref} & ۰ \\ -\dot{\theta}^{ref} & ۰ & v^{ref}(t) \\ ۰ & ۰ & ۰ \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} + \begin{bmatrix} \frac{۱-i_l}{۲} & \frac{۱-i_r}{۲} \\ ۰ & ۰ \\ \frac{۱-i_l}{h} & -\frac{۱-i_r}{h} \end{bmatrix} \begin{bmatrix} r{w_l}^{fb} \\ r{w_r}^{fb} \end{bmatrix}$$

برای به دست آوردن ماتریس A باید مقادیر $\dot{\theta}^{ref}$ و v^{ref} را محاسبه کنیم. برای محاسبه این دو پارامتر از روابط موجود در فصل ۲ کتاب *Gonzalez* استفاده می‌کنیم.

داریم:

$$v^{ref}(t) = \sqrt{\dot{x}^{ref}(t)^{۲} + \dot{y}^{ref}(t)^{۲}}$$

$$\dot{\theta^{ref}} = \frac{\dot{x}^{ref}(t)\ddot{y}^{ref}(t) - \dot{y}^{ref}(t)\ddot{x}^{ref}(t)}{\dot{x}^{ref}(t)^{۲} + \dot{y}^{ref}(t)^{۲}}$$

چون حرکت ربات در مسیر دایره‌ای به مرکز ربات و شعاع ۳ متر مد نظر است، x^{ref} و y^{ref} به صورت زیر در نظر گرفته شده است.

$$x^{ref} = ۳\cos(\frac{\pi}{۱۶}t) \qquad \text{و} \qquad y^{ref} = -۳\sin(\frac{\pi}{۱۶}t)$$

در این صورت خواهیم داشت :

$$\dot{x}^{ref} = -\frac{۳\pi}{۱۶}\sin(\frac{\pi}{۱۶}t) \qquad \text{و} \qquad \ddot{x}^{ref} = -۳\left(\frac{\pi}{۱۶}\right)^{۲}\cos\left(\frac{\pi}{۱۶}t\right)$$

$$\dot{y}^{ref} = \frac{-۳\pi}{۱۶}\cos\left(\frac{\pi}{۱۶}t\right) \qquad \text{و} \qquad \ddot{y}^{ref} = ۳\left(\frac{\pi}{۱۶}\right)^{۲}\sin\left(\frac{\pi}{۱۶}t\right)$$

$$v^{ref}(t) = \sqrt{(\frac{۳\pi}{۱۶})^{۲}\sin(\frac{\pi}{۱۶}t)^{۲} + (\frac{۳\pi}{۱۶})^{۲}\cos(\frac{\pi}{۱۶}t)^{۲}} = \frac{۳\pi}{۱۶}$$

$$\dot{\theta}^{ref}(t) = (\frac{۱۶}{۳\pi})^{۲}\left[\left(-۹\left(\frac{\pi}{۱۶}\right)^{۲}\sin\left(\frac{\pi}{۱۶}t\right)^{۲}\right) - \left(۹\left(\frac{\pi}{۱۶}\right)^{۳}\cos\left(\frac{\pi}{۱۶}t\right)^{۲}\right)\right] = -\frac{\pi}{۱۶}$$

بدین ترتیب معادلات حالت سیستم با استفاده از مقادیر به دست آمده $v^{ref}(t)$ و $\dot{\theta}^{ref}$ و در نظر گرفتن $h = ۰.۱۵\text{m}$ و $i_l = i_r = ۰$ به شکل زیر به دست آمده‌اند.

$$\begin{bmatrix}\dot{e_x}\\ \dot{e_y}\\ \dot{e_\theta}\end{bmatrix} = \begin{bmatrix}۰ & -\frac{\pi}{۱۶} & ۰\\ \frac{\pi}{۱۶} & ۰ & \frac{۳\pi}{۱۶}\\ ۰ & ۰ & ۰\end{bmatrix}\begin{bmatrix}e_x\\ e_y\\ e_\theta\end{bmatrix} + \begin{bmatrix}۰.۵ & ۰.۵\\ ۰ & ۰\\ ۶.۶۶ & -۶.۶۶\end{bmatrix}\begin{bmatrix}rw_l{}^{fb}\\ rw_r{}^{fb}\end{bmatrix}$$

پس برای ماتریس حالت، ورودی و خروجی داریم:

$$A = \begin{bmatrix} ۰ & -\frac{\pi}{۱۶} & ۰ \\ \frac{\pi}{۱۶} & ۰ & \frac{۳\pi}{۱۶} \\ ۰ & ۰ & ۰ \end{bmatrix} \quad , \quad B = \begin{bmatrix} \frac{۱}{۲} & \frac{۱}{۲} \\ ۰ & ۰ \\ \frac{۱}{۰.۱۵} & -\frac{۱}{۰.۱۵} \end{bmatrix} \quad , \quad c = \begin{bmatrix} ۱ & ۰ & ۰ \\ ۰ & ۱ & ۰ \\ ۰ & ۰ & ۱ \end{bmatrix}$$

طبق نتایج شبیه سازی مقاله‌ای که از آن در این پروژه استفاده شده است محدوده معتبر پارامترها برای سیستم خطی سازی شده، به صورت زیر در نظر گرفته شده است:

$$-۲۰\ {rad}/{s} \le w_l \text{ و } w_r \le ۲۰\ {rad}/{s}$$

$$-۰.۵m \le e_x \text{ و } e_y \le ۰.۵m$$

$$-۰.۵rad \le e_\theta \le ۰.۵rad$$

قطری‌سازی و ماتریس تبدیل

برای محاسبه فرم قطری بلوکی از دستور Jordan در متلب استفاده می‌کنیم :

```
J = 3×3 complex
     0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
     0.0000 + 0.0000i   0.0000 - 0.1963i   0.0000 + 0.0000i
     0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.1963i
```

همچنین ماتریس تبدیل تشابهی که استفاده شده است را با همان دستور Jordan بدست می‌آوریم :

```
T1 = 3×3 complex
    -3.0000 + 0.0000i   0.0000 - 1.0000i   0.0000 + 1.0000i
     0.0000 + 0.0000i   1.0000 + 0.0000i   1.0000 + 0.0000i
     1.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
```

تابع تبدیل و نمودار صفر و قطب

تابع تبدیل سیستم بصورت زیر است :

$$G=\begin{bmatrix} \frac{۱۲۸s}{۲۵۶s^{۲}+\pi^{۲}}-\frac{۲۰\pi^{۲}}{۲۵۶s^{۳}+\pi^{۲}s} & \frac{۱۲۸s}{۲۵۶s^{۲}+\pi^{۲}}+\frac{۲۰\pi^{۲}}{۲۵۶s^{۳}+\pi^{۲}s} \\ \frac{۳۲۸\pi}{۲۵۶s^{۲}+\pi^{۲}} & -\frac{۳۱۲\pi}{۲۵۶s^{۲}+\pi^{۲}} \\ \frac{۲۰}{۳s} & -\frac{۲۰}{۳s} \end{bmatrix}$$

حال برای یافتن قطب‌ها و صفرهای انتقال سیستم ابتدا ماینورهای مرتبه اول و دوم سیستم نوشته شده است و سپس چند جمله‌ای مشخصه ماتریس تبدیل به دست آمده است.

ماینورهای مرتبه اول:

$$\frac{۱۲۸s^{۲}+۲۰\pi^{۲}}{۲۵۶s^{۳}+\pi^{۲}s},\ \frac{۱۲۸s^{۲}-۲۰\pi^{۲}}{۲۵۶s^{۳}+\pi^{۲}s},\ \frac{۳۲۸\pi}{۲۵۶s^{۲}+\pi^{۲}},\ \frac{-۳۱۲\pi}{۲۵۶s^{۲}+\pi^{۲}},\ \frac{۲۰}{۳s},\ \frac{-۲۰}{۳s}$$

ماینورهای مرتبه دوم:

$$\frac{-۳۲۰\pi}{۲۵۶s^{۳}+\pi^{۲}s},\quad \frac{۵۱۲۰}{۳(۲۵۶s^{۲}+\pi^{۲})},\quad \frac{-۳۲۰}{۳(۲۵۶s^{۳}+\pi^{۲}s)}$$

معادله مشخصه :

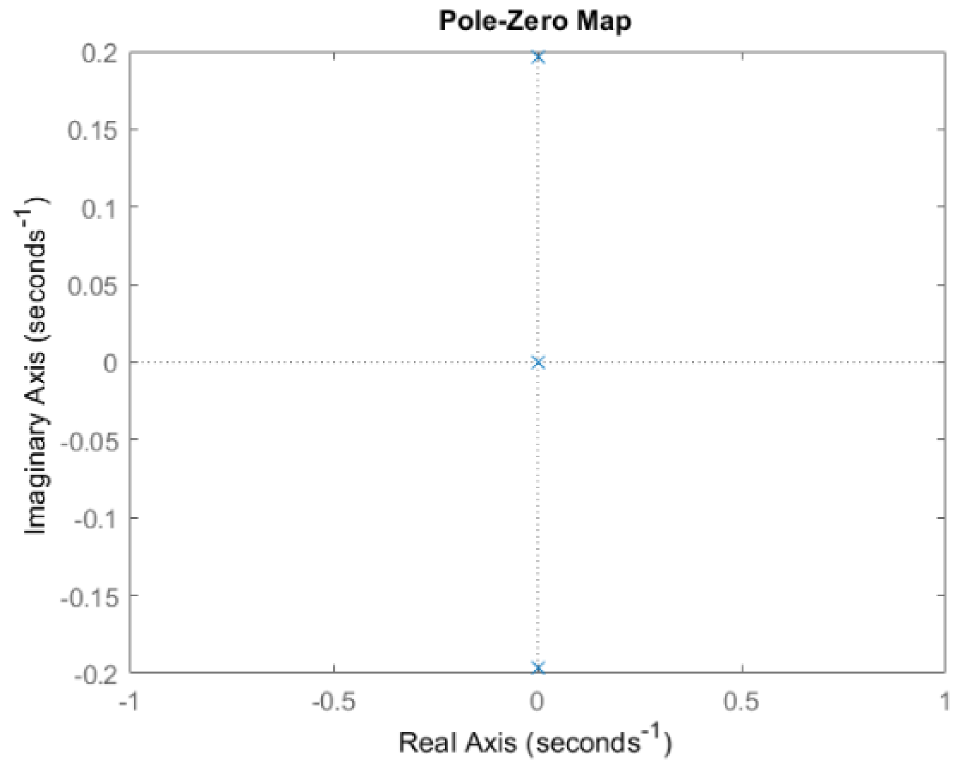
$$p(s)=۳(۲۵۶s^{۳}+\pi^{۲}s)$$

قطب‌های سیستم به شکل زیر می‌باشد:

$$p_{۱}=۰,p_{۲}=\frac{\pi}{۱۶}j,p_{۲}=-\frac{\pi}{۱۶}j$$

برای یافتن صفرهای انتقال لازم است رتبه نرمال سیستم را بدانیم که رتبه نرمال ماتریس تبدیل این سیستم ۲ است. ماینورهای مرتبه ۲ ماتریس تبدیل را در نظر گرفته و مخرج تمامی آن‌ها را معادله مشخصه قرار می‌دهیم در این صورت مشاهده می‌کنیم صورت کسرهای این ماینورها هیچ ب.م.م ای ندارد در نتجیه سیستم صفر انتقال ندارد.

نمودار صفر و قطب سیستم:



پایداری داخلی

برای بررسی پایداری داخلی سیستم لازم است که به رفتار داخلی سیستم و متغیرهای حالت توجه شود. برای تحلیل پایداری داخلی سیستم، در این پروژه از روش اول لیاپانوف استفاده شده است به این صورت که ابتدا نقطه تعادل سیستم غیر خطی به دست آمده است و سپس سیستم غیر خطی حول نقطه تعادل‌اش خطی سازی شده است. که توضیح مفصل این دو بخش در قسمت‌های قبل آمده است. حال باید پایداری سیستم خطی سازی شده(حول نقطه تعادل) بررسی و تحلیل شود برای اینکار با در نظر گرفتن ورودی ۰، اثر شرایط اولیه را روی پاسخ حالت سیستم می‌بینیم. برای اینکار لازم است مقادیر ویژه ماتریس حالت سیستم خطی سازی شده بررسی شوند.

مقادیر ویژه ماتریس حالت:

$$\lambda_{۱} = ۰ \qquad , \qquad \lambda_{۲} = +۰.۱۹۶۳j \qquad , \qquad \lambda_{۳} = -۰.۱۹۶۳j$$

پایداری داخلی به دو بخش پایداری مجانی و پایداری به مفهوم لیاپانف تقسیم می‌شود.

در پایداری مجانبی لازم است قسمت‌های حقیقی تمامی مقادیر ویژه ماتریس حالت اکیدا منفی باشند چون در سیستم مورد مطالعه در این پروژه مقادیر ویژه روی محور موهومی قرار دارند سیستم پایداری مجانبی ندارد.

در مورد پایداری به مفهوم لیاپانف یا پایداری مرزی باید در نظر گرفت که سیستم نباید مقادیر ویژه ای با قسمت حقیقی مثبت داشته باشد و همچنین مقادیر ویژه روی محور در صورتی که تکراری بودند، تکرر جبری و هندسی آن‌ها برابر باشد. که هردو شرط در این سیستم صدق می‌کند پس می‌توان گفت سیستم پایداری مرزی دارد.

در نهایت سیستم پایداری داخلی از نوع مرزی دارد.

پایداری ورودی-خروجی

برای بررسی پایداری ورودی-خروجی باید به قطب‌های ماتریس تبدیل توجه شود. در صورتی که تمامی قطب ها دارای بخش حقیقی اکیدا منفی باشند سیستم دارای پایداری ورودی-خروجی است.

قطب‌های ماتریس تبدیل سیستم:

$$p_{۱} = ۰ \quad , \quad p_{۲} = \frac{\pi}{۱۶}j \quad , \quad p_{۲} = -\frac{\pi}{۱۶}j$$

همان‌طور که قابل مشاهده است قطب‌ها روی محور موهومی هستند پس سیستم دارای پایداری ورودی-خروجی نمی‌باشد.

در بخش قبل دیدیم که سیستم دارای پایداری داخلی بود و در این بخش دیدیم که سیستم پایداری ورودی-خروجی نداشت پس می‌توان گفت اگر سیستم پایدار داخلی باشد نمی‌شود نتیجه گرفت که لزوما پایدار ورودی-خروجی است. فقط در صورتی می‌توان این نتیجه را گرفت که سیستم پایدار داخلی از نوع مجانبی باشد.

پاسخ سیستم

ماتریس انتقال حالت سیستم با دستور expm در متلب به صورت زیر می‌باشد:

$$e^{At} = \begin{bmatrix} \frac{e^{\frac{-\pi t j}{۱۶}}}{۲} + \frac{e^{\frac{\pi t j}{۱۶}}}{۲} & \frac{-e^{\frac{-\pi t j}{۱۶}}j}{۲} + \frac{e^{\frac{\pi t j}{۱۶}}j}{۲} & \frac{۳e^{\frac{-\pi t j}{۱۶}}}{۲} + \frac{۳e^{\frac{\pi t j}{۱۶}}}{۲} - ۳ \\ \frac{e^{\frac{-\pi t j}{۱۶}}j}{۲} - \frac{e^{\frac{\pi t j}{۱۶}}j}{۲} & \frac{e^{\frac{-\pi t j}{۱۶}}}{۲} + \frac{e^{\frac{\pi t j}{۱۶}}}{۲} & \frac{۳e^{\frac{-\pi t j}{۱۶}}j}{۲} - \frac{۳e^{\frac{\pi t j}{۱۶}}j}{۲} \\ ۰ & ۰ & ۱ \end{bmatrix}$$

پاسخ حالت سیستم به ازای شرایط اولیه $x_. = \begin{bmatrix} -۰.۱ \\ -۰.۲ \\ -۰.۲ \end{bmatrix}$ با استفاده از فرمول $x = e^{At}x_.$ به شکل زیر به دست آمده است:

$$x(t_.) = \begin{bmatrix} \frac{۳}{۵} + e^{-\frac{\pi tj}{۱۶}}\left(\frac{-۷}{۲۰} + \frac{۱}{۱۰}j\right) + e^{\frac{\pi tj}{۱۶}}\left(\frac{-۷}{۲۰} - \frac{۱}{۱۰}j\right) \\ e^{-\frac{\pi tj}{۱۶}}\left(\frac{-۱}{۱۰} - \frac{۷}{۲۰}j\right) + e^{\frac{\pi tj}{۱۶}}\left(\frac{-۱}{۱۰} + \frac{۷}{۲۰}j\right) \\ -\frac{۱}{۵} \end{bmatrix}$$

پاسخ خروجی سیستم به ازای شرایط اولیه قبلی و ورودی پله واحد حاصل جمع پاسخ ورودی صفر و پاسخ شرایط اولیه صفر است.

پاسخ شرایط اولیه صفر سیستم:

$$\begin{bmatrix} \delta(t) \\ -\frac{\pi}{۱۶} \\ ۰ \end{bmatrix}$$

پاسخ کامل خروجی به شکل زیر می‌باشد:

$$y(t) = \begin{bmatrix} \delta(t) + \frac{۳}{۵} + e^{-\frac{\pi tj}{۱۶}}\left(\frac{-۷}{۲۰} + \frac{۱}{۱۰}j\right) + e^{\frac{\pi tj}{۱۶}}\left(\frac{-۷}{۲۰} - \frac{۱}{۱۰}j\right) \\ -\frac{\pi}{۱۶} + e^{-\frac{\pi tj}{۱۶}}\left(\frac{-۱}{۱۰} - \frac{۷}{۲۰}j\right) + e^{\frac{\pi tj}{۱۶}}\left(\frac{-۱}{۱۰} + \frac{۷}{۲۰}j\right) \\ -\frac{۱}{۵} \end{bmatrix}$$

تحریک فرکانس خاص

برای اینکه مود خاصی در خروجی تحریک نشود کافی است شرایط اولیه را به طوری تعیین کنیم که ترکیب خطی از بردار ویژه های متناسب با مقدار ویژه های مود های دیگر سیستم باشد.
در مراحل قبل با استفاده از دستور Jordan علاوه بر فرم جردن ماتریس حالت سیستم ، ماتریس تبدیل تشابهی که ماتریس اصلی ما را به فرم جردن درمی‌آورد نیز بدست آمد که ستون های آن همان بردار ویژه های ماتریس حالت است .

T1 = 3×3 complex

	1	2	3
1	-3.0000 + 0.0000i	0.0000 - 1.0000i	0.0000 + 1.0000i
2	0.0000 + 0.0000i	1.0000 + 0.0000i	1.0000 + 0.0000i
3	1.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i

J = 3×3 complex

	1	2	3
1	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.0000i
2	0.0000 + 0.0000i	0.0000 - 0.1963i	0.0000 + 0.0000i
3	0.0000 + 0.0000i	0.0000 + 0.0000i	0.0000 + 0.1963i

حال شرایط اولیه را ترکیب خطی ستون دوم و سوم در نظر می‌گیریم. با این فرض مقدار ویژه صفر $\lambda_۱ = 0$ در خروجی تحریک نمی‌شود. برای اطمینان از متلب استفاده می‌کنیم :

$$x_۱ = \left(۰ \times \begin{bmatrix} -۳ \\ ۰ \\ +۱ \end{bmatrix} \right) + \left(۱ \times \begin{bmatrix} ۰ - ۱j \\ ۱ + ۰j \\ ۰ \end{bmatrix} \right) + \left(۱ \times \begin{bmatrix} ۰ + ۱j \\ ۱ + ۰j \\ ۰ \end{bmatrix} \right) = \begin{bmatrix} ۰ \\ ۲ \\ ۰ \end{bmatrix}$$

$$x_۱(t0) = C_e * e^{At} * x_۱ = \begin{bmatrix} -e^{-\frac{\pi t j}{۱۶}} j + e^{\frac{\pi t j}{۱۶}} j \\ e^{-\frac{\pi t j}{۱۶}} + e^{\frac{\pi t j}{۱۶}} \\ ۰ \end{bmatrix}$$

تجزیه کالمن

ابتدا برای بررسی کنترل پذیری و رویت پذیری سیستم ماتریس‌های کنترل پذیری و رویت پذیری سیستم تشکیل داده شده است و رنک آن‌ها محاسبه شده است.

```
Co = 3×6
```

	1	2	3	4	5	6
1	0.5000	0.5000	0	0	-0.7903	0.7518
2	0	0	4.0252	-3.8288	0	0
3	6.6667	-6.6667	0	0	0	0

$$C_o = \begin{bmatrix} ۰.۵ & ۰.۵ & ۰ & ۰ & -۰.۷۹۰۳ & ۰.۷۵۱۸ \\ ۰ & ۰ & ۴.۰۲۵۲ & -۳.۸۲۸۸ & ۰ & ۰ \\ ۶.۶۶۶۷ & -۶.۶۶۶۷ & ۰ & ۰ & ۰ & ۰ \end{bmatrix}$$

```
Ob = obsv(A_e,C_e)

Ob = 9×3
    1.0000         0         0
         0    1.0000         0
         0         0    1.0000
         0   -0.1963         0
    0.1963         0    0.5890
         0         0         0
   -0.0386         0   -0.1157
         0   -0.0386         0
         0         0         0
```

$$O_b = \begin{bmatrix} ۱ & ۰ & ۰ \\ ۰ & ۱ & ۰ \\ ۰ & ۰ & ۱ \\ ۰ & -۰.۱۹۶۳ & ۰ \\ ۰.۱۹۶۳ & ۰ & ۰.۵۸۹۰ \\ ۰ & ۰ & ۰ \\ -۰.۰۳۸۶ & ۰ & -۰.۱۱۵۷ \\ ۰ & -۰.۰۳۸۶ & ۰ \\ ۰ & ۰ & ۰ \end{bmatrix}$$

همان طور که در شکل بالا مشخص شده است رنک سطری ماتریس کنترل پذیری و رنک ستونی ماتریس رویت پذیری سه (کامل) می‌باشد پس سیستم هم کنترل پذیر و رویت پذیر است.

پس سیستم فقط دارای یک زیر سیستم رویت پذیر و کنترل پذیر به شکل زیر می‌باشد.

$$Abar = \begin{bmatrix} ۰ & -۰.۱۹۶۳ & -۰.۵۸۹۰ \\ ۰.۱۹۶۳ & ۰ & ۰ \\ ۰ & ۰ & ۰ \end{bmatrix} \qquad Bbar = \begin{bmatrix} ۰ & ۰ \\ -۰.۵ & -۰.۵ \\ -۶.۶۶۶۷ & ۶.۶۶۶۷ \end{bmatrix}$$

$$Cbar = \begin{bmatrix} ۰ & -۱ & ۰ \\ ۱ & ۰ & ۰ \\ ۰ & ۰ & -۱ \end{bmatrix}$$

تحقق مینیمال

با توجه به کامل بودن رنک ماتریس های کنترل پذیری و رویت پذیری، همه ی مود های سیستم کنترل پذیر و رویت پذیر می‌باشند. می‌دانیم در تابع تبدیل کل سیستم مودهایی را خواهیم دید که هم کنترل پذیر و هم رویت باشند. اگر سیستم قابل تبدیل به زیر سیستم های کنترل پذیر-رویت پذیر، کنترل ناپذیر-رویت پذیر،کنترل پذیر-رویت ناپذیر و کنترل ناپذیر-رویت ناپذیر باشد، تابع تبدیل کل سیستم معادل با تابع تبدیل زیر سیستم کنترل پذیر-رویت پذیر است. از آنجایی که سیستم ما فقط زیر سیستم کنترل پذیر-رویت پذیر دارد، تحقق فضای حالت برای این تابع تبدیل مینیمال می‌باشد.

فاز دوم

زیر سیستم کاهش ناپذیر

سیستم مورد مطالعه در این پروژه همان‌طور که در بخش قبل بررسی شد مینیمال است یعنی خودش کاهش ناپذیر است.

طراحی فیدبک حالت

شرایط مطلوب در نظر گرفته شده به شرح زیر می‌باشد

$$P.O < ۴\%$$

$$Ts < ۰.۵s$$

با توجه به روابط اورشوت و زمان نشست قطب‌های مطلوب را به دست می‌آوریم.

$$P.O = ۱۰۰ e^{\frac{-\pi\zeta}{\sqrt{۱-\zeta^۲}}} \rightarrow \zeta = ۰.۴$$

$$Ts = \frac{۴}{\zeta\omega_n} = ۲ \rightarrow \omega_n = ۵$$

$$desired\ poles = -۱ \pm j4.۸۹۹$$

تا اینجا برای یافتن قطب‌های مطلوب سیستم را با سیستم درجه دو استاندارد تقریب زدیم تا بتوانیم از روابط زمان نشست و اورشوت سیستم درجه دو استاندارد استفاده کنیم. حال باید یک قطب را با نسبت زیادی نسبت به ۳۰- دورتر از محور $j\omega$ در نظر گرفت تا بتوان معادله مشخصه مطلوب را محاسبه کرد.

معادله مشخصه مطلوب به شکل زیر می‌شود:

$$\alpha(\mathrm{s}) = s^{۳} + ۳۲s^{۲} + ۸۵s + ۷۵۰$$

برای دست یابی به ماتریس K لازم است که دو ماتریس F و $\overline{K}$ را به طوری تعیین کنیم که زوج $(F, \overline{K})$ روئت پذیر باشند.

ماتریس F و $\overline{K}$ به شکل زیر در نظر گرفته شده‌اند:

$$F = \begin{bmatrix} -۱ & ۴.۸۹۹ & ۰ \\ -۴.۸۹۹ & -۱ & ۰ \\ ۰ & ۰ & -۳۰ \end{bmatrix} \quad \overline{K} = \begin{bmatrix} ۱ & ۰ & ۰ \\ ۰ & ۰ & ۱ \end{bmatrix}$$

$$rank\big(obsv(F, K)\big) = ۳$$

رنک ماتریس روئت پذیری کامل است پس ماتریس‌ها به درستی انتخاب شده اند.

حال برای به دست آوردن ماتریس K معادله لیاپانوف $AP - PF = B\overline{K}$ باید حل شود.

$$P = lyap(A, -F, -B\overline{K}) = \begin{bmatrix} ۰.۰۲۳۶ & ۰.۱۰۳۲ & ۰.۰۱۶۷ \\ ۰.۱۴۸۳ & -۰.۰۶۳۳ & ۰.۰۰۴۳ \\ ۰.۲۶۶۷ & ۱.۳۰۶۴ & -۰.۲۲۲۲ \end{bmatrix}$$

ماتریس نهایی K از رابطه ی زیر به دست آمده است.

$$K = \overline{K}P^{-۱} = \begin{bmatrix} ۱.۱۷۵۶ & ۶.۱۸۴۴ & ۰.۲۰۶۷ \\ ۲۹.۱۱۲۴ & -۰.۴۵۸۰ & -۲.۳۲۱۷ \end{bmatrix}$$

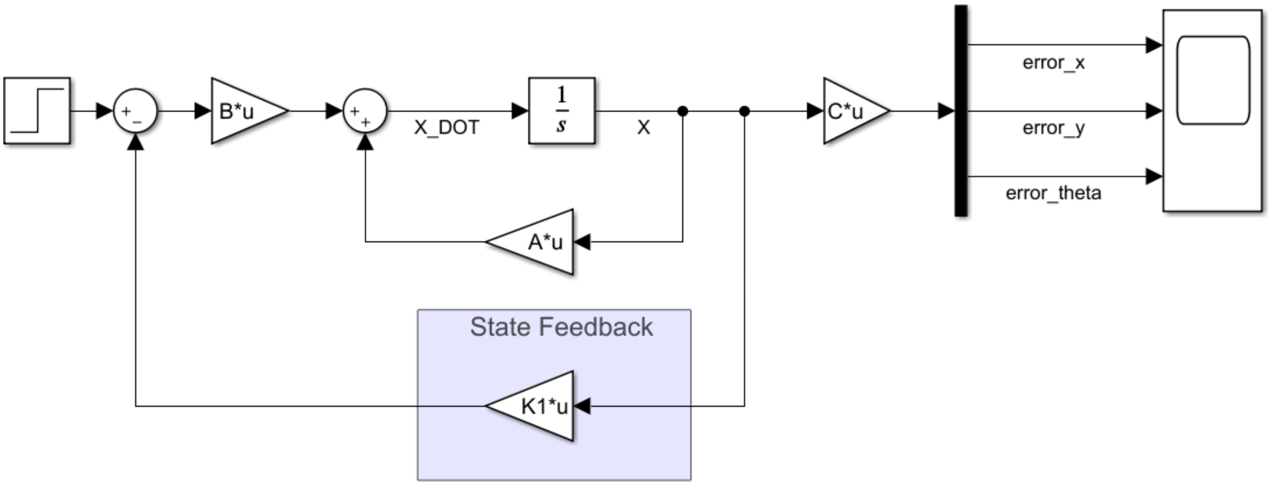
همچنین در متلب با استفاده از دستور place، K به شکل زیر به دست آمده است.

$$K = \begin{bmatrix} ۲۶.۶۸۳۱ & ۱۴.۲۴۵۸ & -۸.۰۵۳۷ \\ ۲۷.۹۰۴۳ & ۸.۴۵۸۸ & -۸.۷۵۹۶ \end{bmatrix}$$

با مقایسه نتایج شبیه سازی با دو بهره فیدبک حالت به دست آمده در بالا، دومی نتایج بهتری داشت و در نتیجه برای ادامه مراحل از آن استفاده شده است.

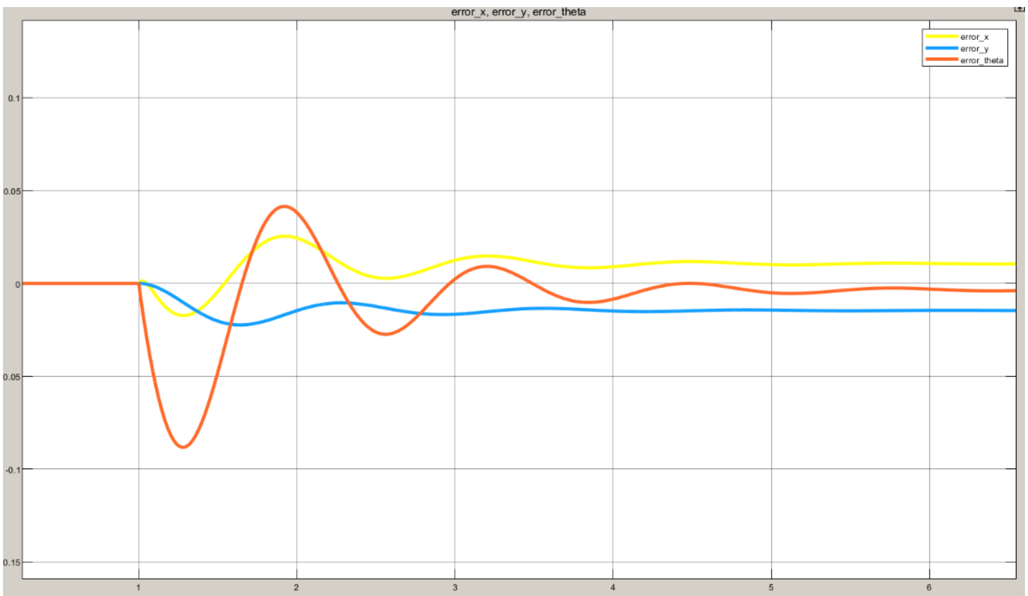
تحلیل سیستم خطی و غیرخطی با فیدبک حالت

برای شبیه سازی سیستم خطی با فیدبک حالت طراحی شده در قسمت قبل سیستم با فیدبک حالت به شکل زیر در سیمولینک پیاده سازی شده است.



انتظار این بود که با جابابی قطب‌های حلقه بسته در مکان‌های مطلوب، حالت‌ها به پایداری برسند.

نتیجه شبیه سازی:



حال فیدبک حالت طراحی شده برای سیستم خطی را بر روی سیستم غیرخطی امتحان می‌کنیم.

برای این منظور سیستم غیرخطی را به صورت زیر طراحی کردیم:

```
function e_dot  = fcn(e,U)
e_dot = [0.1;-0.1;0.05];
theta_dot_ref = -(pi/16) ;
v_ref = 3*(pi/16) ;
e_dot(1)=(theta_dot_ref - U(1))*e(2) + v_ref*cos(e(3))+(U(2) - v_ref);
e_dot(2)=(U(1)-theta_dot_ref)*e(1) + v_ref*sin(e(3));
e_dot(3)= U(1);
```

که این سسیتم از معادلات غیرخطی زیر بدستآمدهاند:

$$\dot{e_x} = \dot{\theta} e_y + v^{ref}\cos(\theta^{ref} - \theta) - v$$

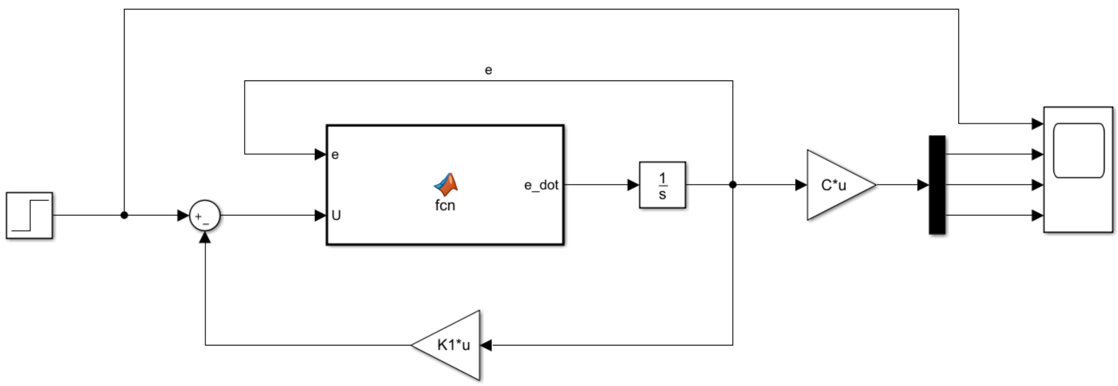
$$\dot{e_y} = -\dot{\theta} e_x + v^{ref}\sin(\theta^{ref} - \theta)$$

$$\dot{e_\theta} = \dot{\theta}^{ref} - \dot{\theta}$$

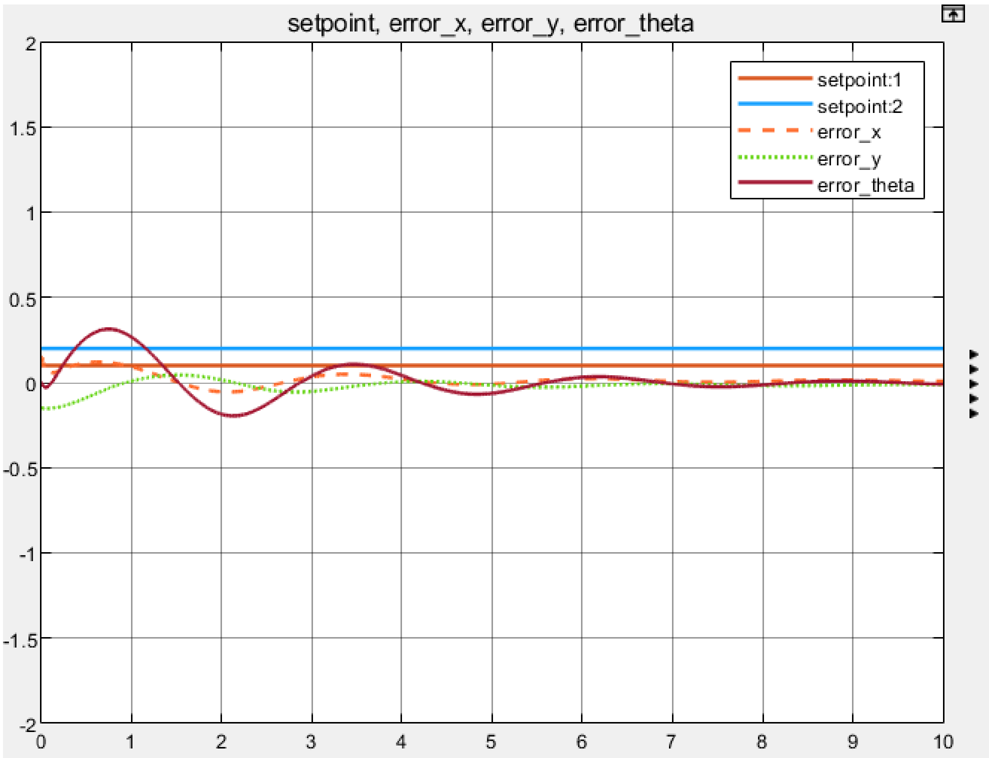
ورودی سیستم، سرعت زاویه‌ای چرخ‌های چپ و راست هستند که توسط ماتریس تبدیل زیر از سرعت‌های خطی و زاویه‌ای سیستم به دست آمده‌اند.

$$\begin{bmatrix} \omega_f \\ V_f \end{bmatrix} = \begin{bmatrix} r/h & -r/h \\ r/2 & r/2 \end{bmatrix} \begin{bmatrix} \omega_l \\ \omega_r \end{bmatrix}$$

فیدبک حالت طراحی شده در قسمت قبل به شکل به سیستم غیرخطی اعمال شده است.



خروجی سیستم غیر خطی با فیدبک حالت و با شرایط اولیه به صورت زیر می‌باشد:



با در نظر گرفتن شرایط اولیه [0.15; -0.15;0] که در محدوده مجاز برای متغیرهای حالت می‌باشد نمودار خطاها به صورت بالا درآمده است. همان‌طور که قابل مشاهده است مقدار خطا در مقایسه با شعاع دایره که ۳ متر می‌باشد بسیار ناچیز است و به صفر میل می‌کند.

بررسی ناپایداری سیستم غیرخطی با فیدبک حالت

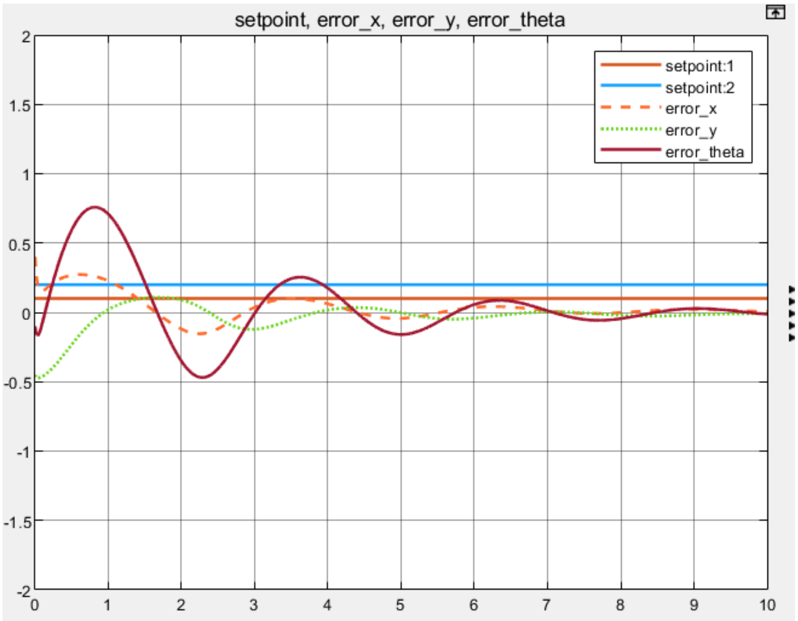
با در نظر گرفتن محدوده مجازی که مقاله مرجع بیان کرده است، محدوده جذب برای سیستم مورد مطالعه به شرح زیر می‌باشد

$-۲۰\ {rad}/{s} \leq w_l$ و $w_r \leq ۲۰\ {rad}/{s}$

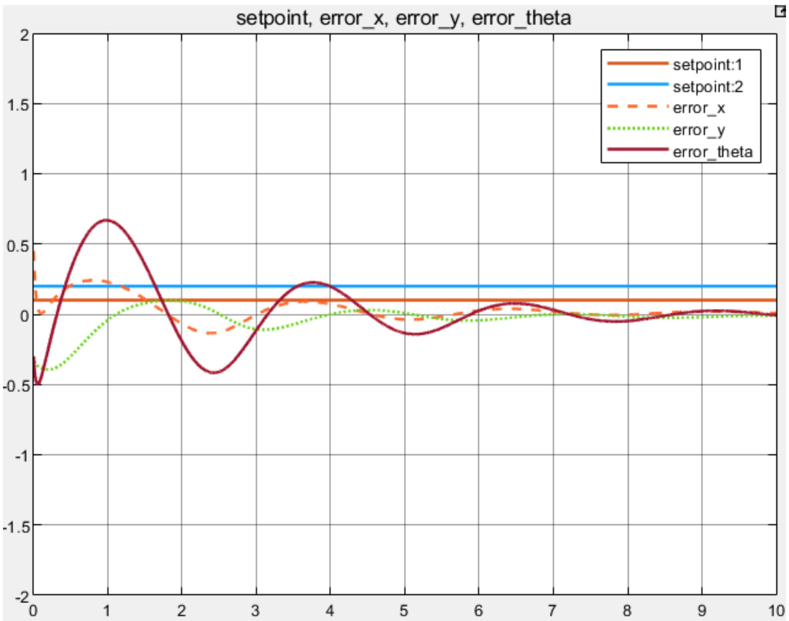
$-۰.۵m \leq e_x$ و $e_y \leq ۰.۵m$

$-۰.۵rad \leq e_\theta \leq ۰.۵rad$

با توجه به محدوده بالا چند شرط اولیه بررسی شده است:



شرایط اولیه در نظر گرفته شده [0.4; -0.45; -0.1]



شرایط اولیه در نظر گرفته شده [0.45; -0.3; 0.3]

با اعمال شرایط اولیه مختلف دریافتیم که به ازای برخی شرایط اولیه که در محدوده تعیین شده توسط مقاله بود، سیستم ناپایدار شد در نتیجه برای حل مشکل ناپایداری محدوده جذب باید کوچکتر انتخاب شود.

$-۰.۲۵m \leq e_x$ و $e_y \leq ۰.۲۵m$

$-۰.۲rad \leq e_\theta \leq ۰.۲rad$

طراحی پیش‌جبرانساز استاتیکی و دینامیکی

برای دنبال کردن ورودی مرجع لازم است یک جبران‌ساز استاتیکی در سیستم قرار دهیم.

این جبرانساز به صورت زیر محاسبه می‌شود:

$$StaticGain = (C * (A - B * K)^{-۱} * B)^{-۱}$$

همان‌طور که مشاهده می‌شود مقدار این گین همان مقدار تابع تبدیل حلقه بسته سیستم در صفر می‌باشد.

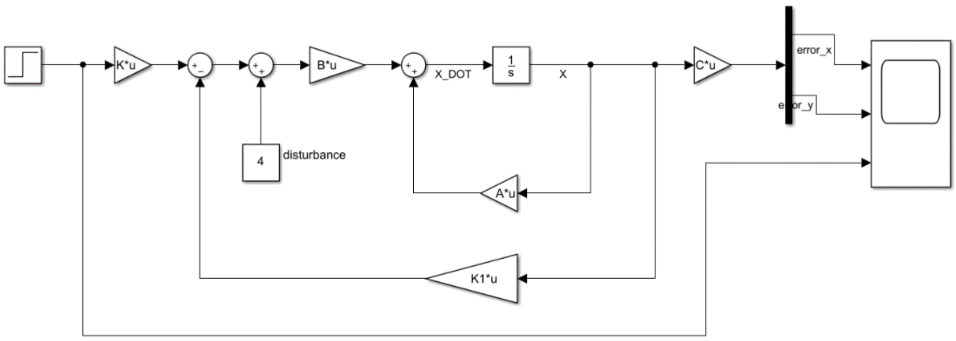
این گین در متلب به صورت زیر محاسبه شده است:

```
%% static Controller
syms s;
G_closeLoop = inv(C*(-(A-B*K1)^-1)*B);
```

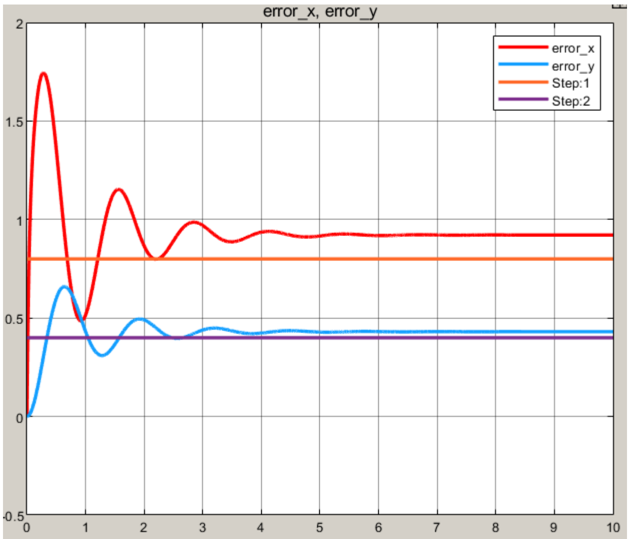
گین به دست آمده به صورت زیر است:

$$G - Closeloop = \begin{bmatrix} ۲۹.۲۵ & ۱۵.۵۹۷۳ \\ ۳۰.۷۵ & ۹.۸۶۷۷ \end{bmatrix}$$

برای شبیه‌سازی، مدار زیر در سیمولینک بسته شده است:



خروجی این سیستم به ازای ورودی مرجع پله و اغتشاش ثابت:



در شکل مشاهده می‌شود که پاسخ خطای حالت دائم دارد و این به دلیل نبود ترم انتگرال گیر می‌باشد.

برای کم‌کردن خطای حالت دائم، از یک جبرانساز دینامیکی استفاده می‌کنیم.

با قرار دادن این گین دو حالت جدید به حالت‌های سیستم ما اضافه می‌شود، ماتریس حالت و ماتریس ورودی جدید به صورت زیر است:

$$\bar{A} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}$$

برای به دست آوردن گین‌های پسخور و پیشخور ابتدا سیستم را با استفاده از ماتریس حالت و ماتریس ورودی جدید تشکیل داده، قطب‌های مطلوب آن را محاسبه کرده سپس با استفاده از دستور $place$ در متلب گین های پسخور و پیشخور را محاسبه کرده ایم.

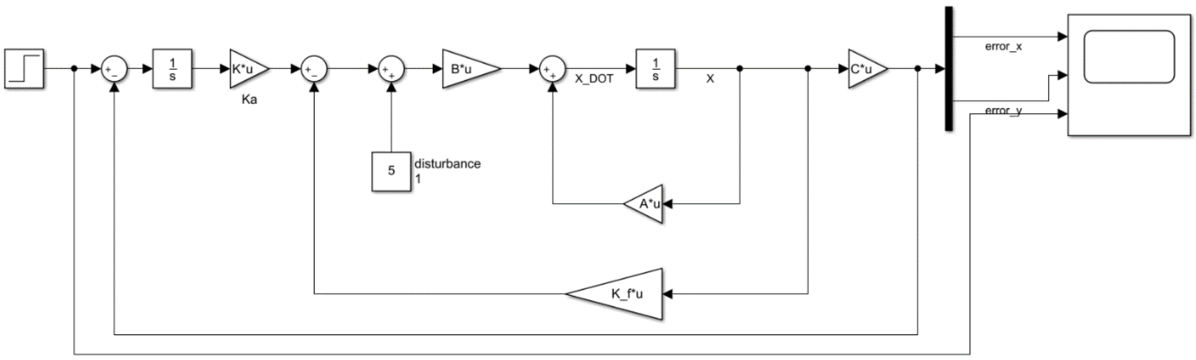
```
A_bar = [0 -pi/16 0 0 0;pi/16 0 3*pi/16 0 0; 0 0 0 0 0; 1 0 0 0 0; 0 1 0 0 0];
B_bar = [1/2 1/2;0 0;1/0.15 -1/0.15; 0 0; 0 0];
C_bar = [1 0 0 0 0; 0 1 0 0 0];
D_bar = [0 0 0 0 0; 0 0 0 0 0];
K_bar = place(A_bar, B_bar, [-60 -55 -70 -1+4.899*i -1-4.899*i ]);
Ka = K_bar(:,4:5);
K_f = K_bar(:,1:3);
```

گین‌های به دست‌آمده به صورت زیر است:

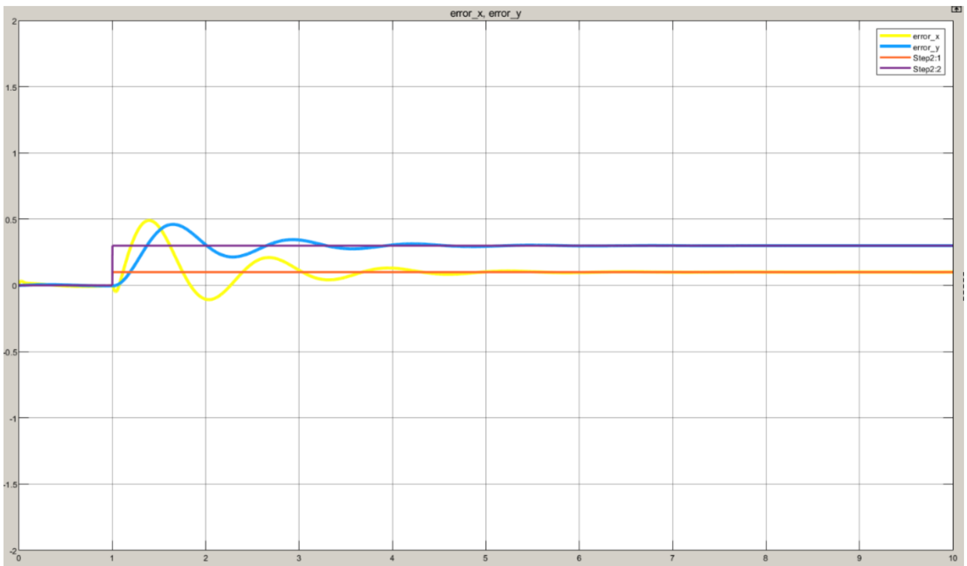
```
K_f =

   1.0e+03 *

    0.1093   -1.1934   -0.0071
    0.1167   -1.4132   -0.0182
```

```
Ka =

   1.0e+03 *

    3.1616   -2.2756
    3.6396   -3.0848
```

برای شبیه‌سازی مدار زیر در سیمولینک بسته شده است:



خروجی سیستم به ازای ورودی مرجع پله و اغتشاش ثابت:

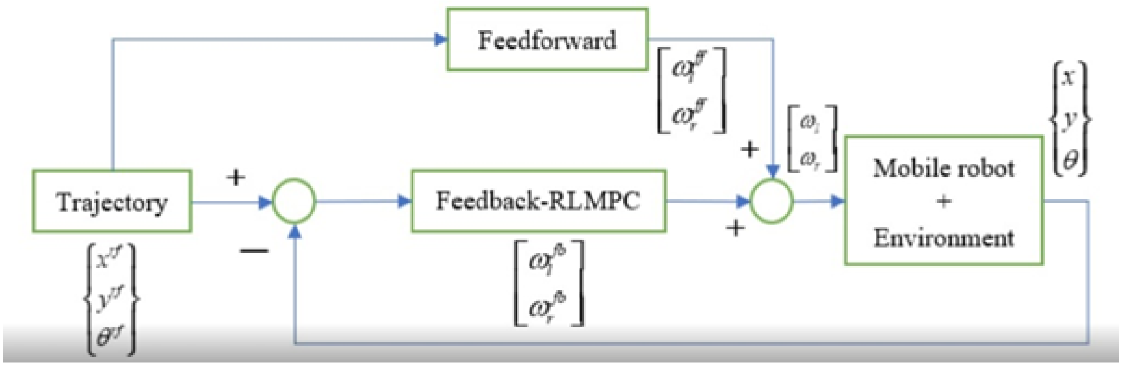


در این حالت مشاهده شد که به ازای ورودی مرجع پله خطا ورودی مرجع را دنبال می‌کند. به همین دلیل نیاز شد که سیستم ارتقا داده شود به طوری که به ازای ورودی مرجع دایره حالت‌های سیستم که همان خطاها هستند به صفر میل کنند.

ارتقا مدل شبیه‌سازی شده سیستم:

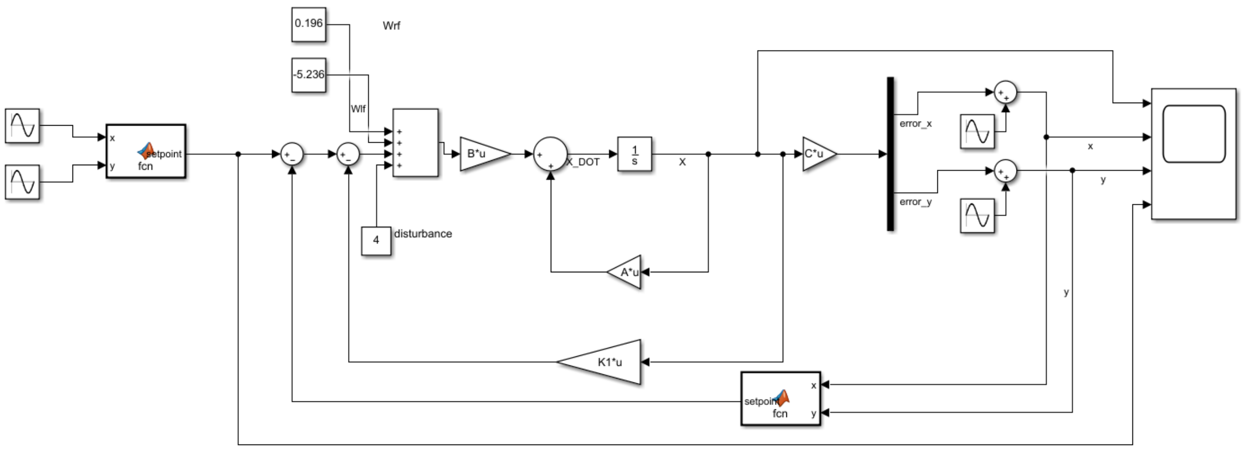
در این بخش، برای مشاهده خروجی شبیه سازی شده mobile robot لازم بود به خروجی فعلی سیستم دسترسی داشته باشیم تا با مقایسه آن با مسیر مطلوب که در این پروژه دایره‌ای به شعاع سه متر در نظر گرفته شده است، به بررسی عملکرد سیستم در ردیابی این مسیر بپردازیم. به این منظور مدل فضای حالت که بر اساس خطا حالت‌های سیستم طراحی شده بود را با تغییراتی به مدل واقعی سیستم برای شبیه سازی تبدیل کردیم. در این مدل ورودی مرجع همان مسیر دایره‌ای مطلوب است و خروجی‌ها موقعیت فعلی(x و y ربات) می‌باشد که انتظار می‌رود خروجی سیستم هم تقریبا همان مسیر دایره‌ای باشد.

برای رسیدن به هدف مذکور از مدل زیر استفاده شد.



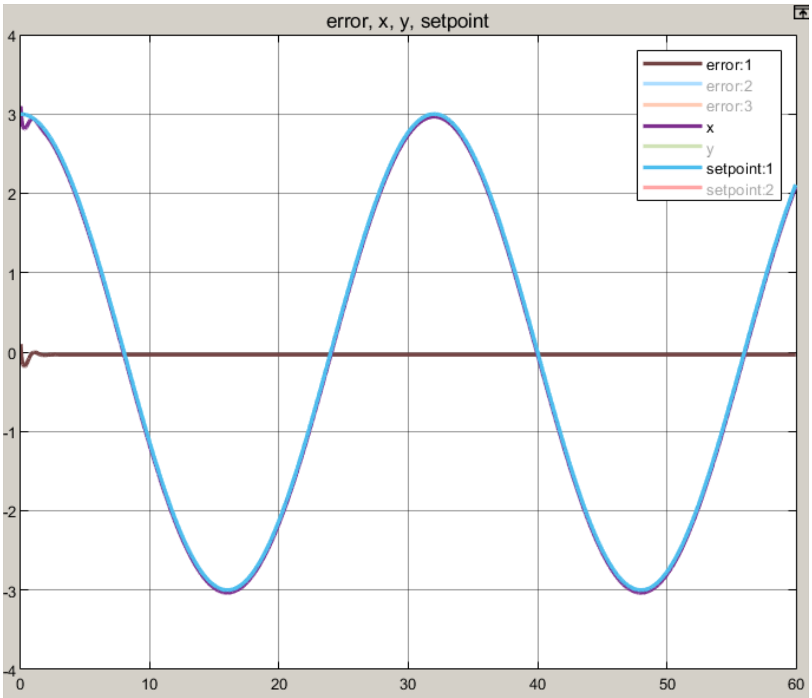
به این منظور یک ترم feedfrorward که سرعت زاویه‌ای چرخ‌های راست و چپ هستند(در قسمت یک محاسبه شده است.) به ورودی سیستم اعمال شد. همچنین خروجی ماتریس حالت سیستم که خطاهای راستای x و y بود با x وy مطلوب که مسیر دایره‌ای را تشکیل می‌دادند جمع شد تا به خروجی‌های فعلی دست یابیم. ورودی مرجع سیستم هم مسیر دایره‌ای مطلوب در نظر گرفته شد.

مدل نهایی به شکل زیر پیاده سازی شده است.

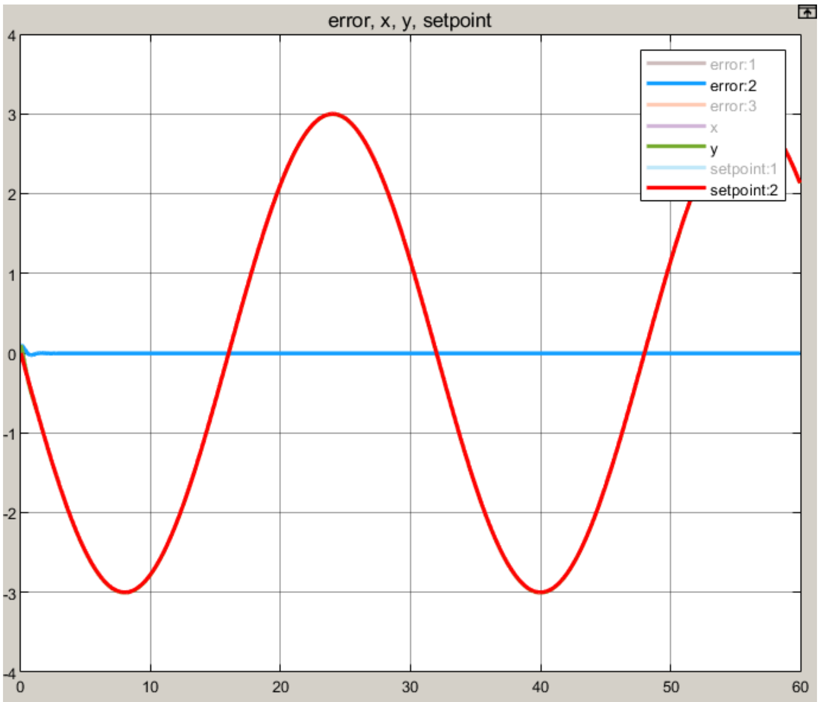


برای مشاهده تصویر با کیفیت بیشتر اینجا کلیک کنید.

نتیجه خروجی و خطا در راستای x به همراه setpoint در زیر آورده شده است.

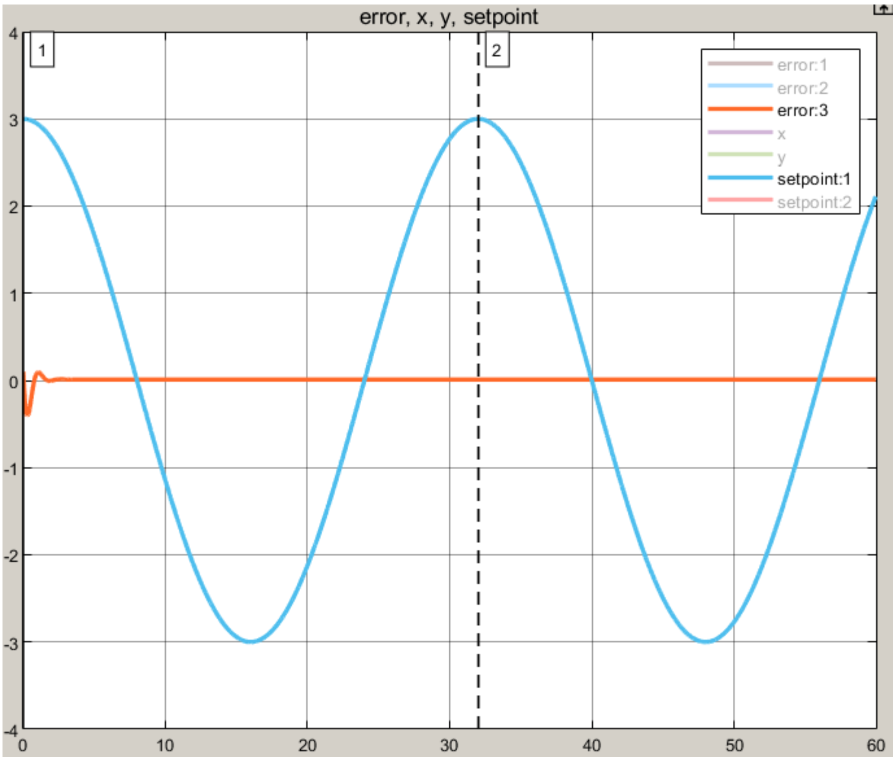


نتیجه خروجی و خطا در راستای y به همراه setpoint در زیر آورده شده است.



تذکر: در نمودار اول سیگنال‌های آبی و بنفش و در نمودار دوم سیگنال‌های قرمز و سبز روی یکدیگر قرار گرفته‌اند که این به معنای ردیابی بسیار خوب سیستم است.

همان‌طور که انتظار می‌رفت با توجه به سرعت زاویه‌ای ربات که برابر $\omega = \frac{\pi}{۱۶}$ می‌باشد، مدت زمان طی کردن یک دور کامل برابر ۳۲ ثانیه می‌باشد.



فاز سوم

طراحی رویتگر مرتبه کامل و رویتگر مرتبه کاهش یافته

برای طراحی رویت‌گر مرتبه کامل قطب‌های رویتگر دورتر از قطب‌های غالب سیستم در نظر گرفته شده‌اند، تا رویتگر سریع‌تر از سیستم اصلی باشد. بدین منظور قطب‌های آن را در[-۲۰, -۲۰, -۱۰۰]

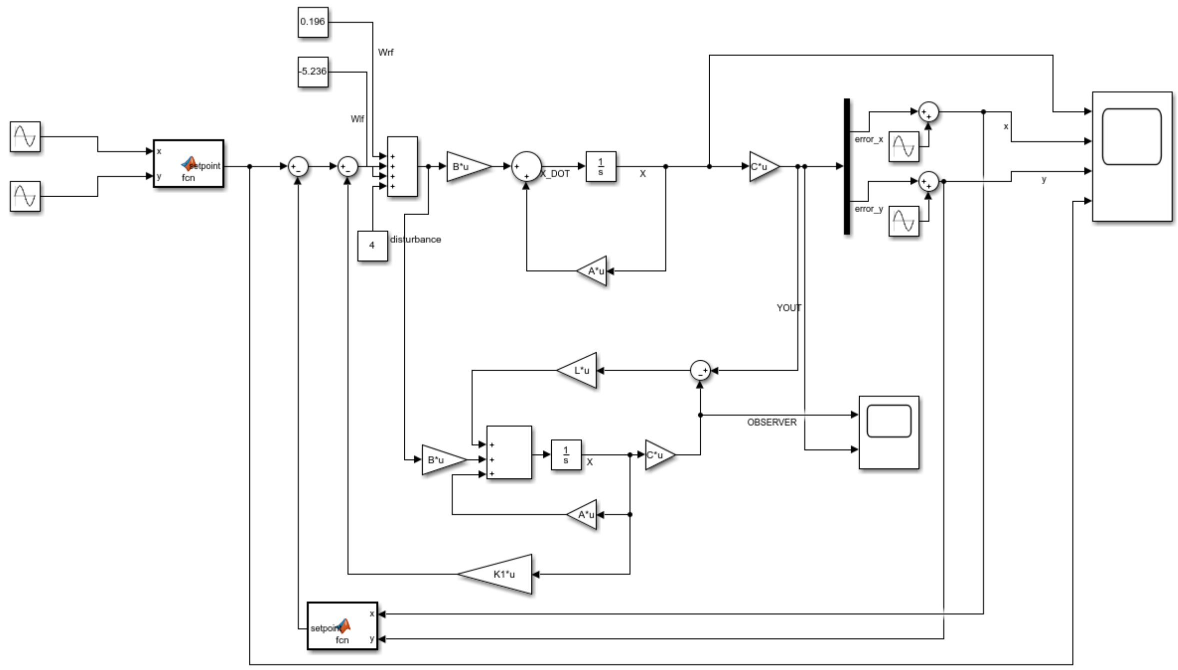
قرار داده شده است. و به شکل زیر در متلب ماتریس L برای رویتگر مرتبه کامل تشکیل داده شده است.

```
%% full-order observer
L = place(A', C', [-20, -20, -100]);
L = L';
```

که در نهایت برای ماتریس L داریم:

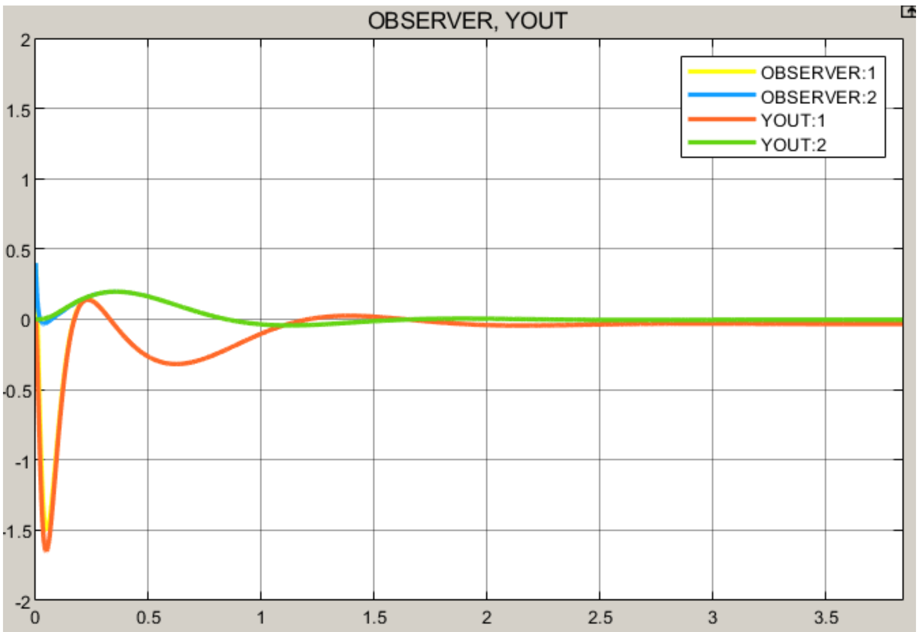
$$L = ۱۰^{۳} \begin{bmatrix} ۰.۰۲ & -۰.۰۰۰۲ \\ ۰.۰۰۰۲ & ۰.۱۲ \\ ۰ & ۳.۳۹۵۳ \end{bmatrix}$$

برای شبیه سازی سیستم با رویتگر مرتبه کامل به شکل زیر عمل شده است.



برای مشاهده تصویر با کیفیت بیشتر اینجا کلیک کنید.

در شکل زیر حالت‌های تخمین زده شده به همراه حالت‌های اصلی سیستم قابل مشاهده می‌باشد.

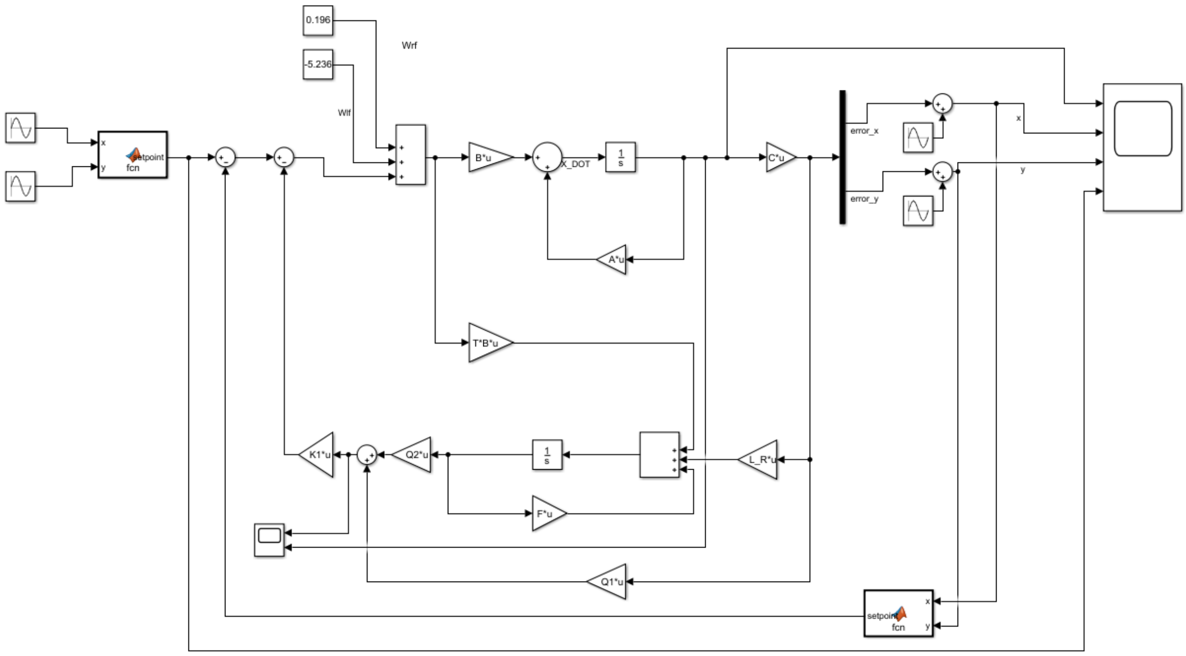


که این نتیجه برای حالتی است که رویتگر شرایط اولیه [۰,۵ ۰,٤ ۰,٤] و سیستم اصلی دارای شرایط اولیه [۰ ۰,۱ ۰,۱] می‌باشد.

همان‌طور که قابل مشاهده است با وجود شرایط اولیه نسبتا متفاوت حالت‌ها به درستی تخمین زده شده اند.

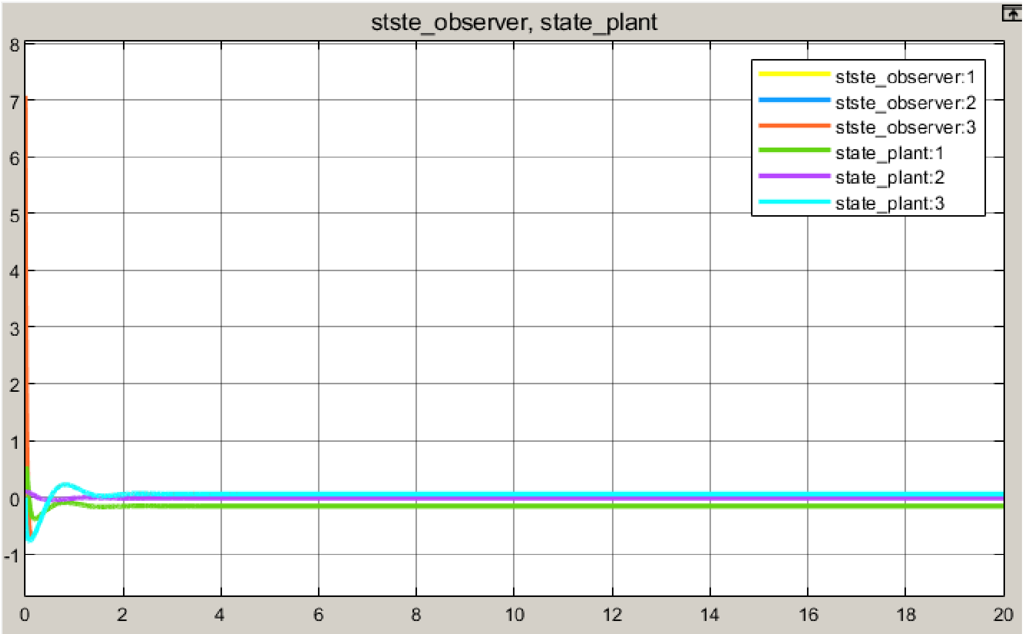
برای طراحی رویتگر مرتبه کاهش یافته فرض شده است که دو حالت سیستم در دسترس هستند و یک حالت سیستم که زاویه می‌باشد نیاز است توسط رویتگر تخمین زده شود. بدین منظور قطب مطلوب رویتگر در ۴۰- در نظر گرفته شده است. بدین منظور رویتگر مرتبه کاهش یافته به شکل زیر طراحی و در سیمولینک پیاده سازی شده است.

```
%% reduced-order observer
F = -40;
L_R = [1 -400];
T = lyap(-F, A, -L_R*C);
Q = inv([1 0 0; 0 1 0; T]);
Q1 = [1 0; 0 1; -0.5031 67.9061];
Q2 = [0 0 6.7909]';
```



برای مشاهده تصویر با کیفیت بیشتر اینجا کلیک کنید.

بررسی عملکرد رویتگر کاهش یافته:



همان‌طور که قابل مشاهده است رویتگر حالت مورد نظر را با دقت خوبی تخمین زده است.

طراحی پیش‌جبرانساز استاتیکی و دینامیکی با استفاده از حالت‌های تخمین زده شده توسط رویتگر مرتبه کامل و رویتگر مرتبه کاهش یافته

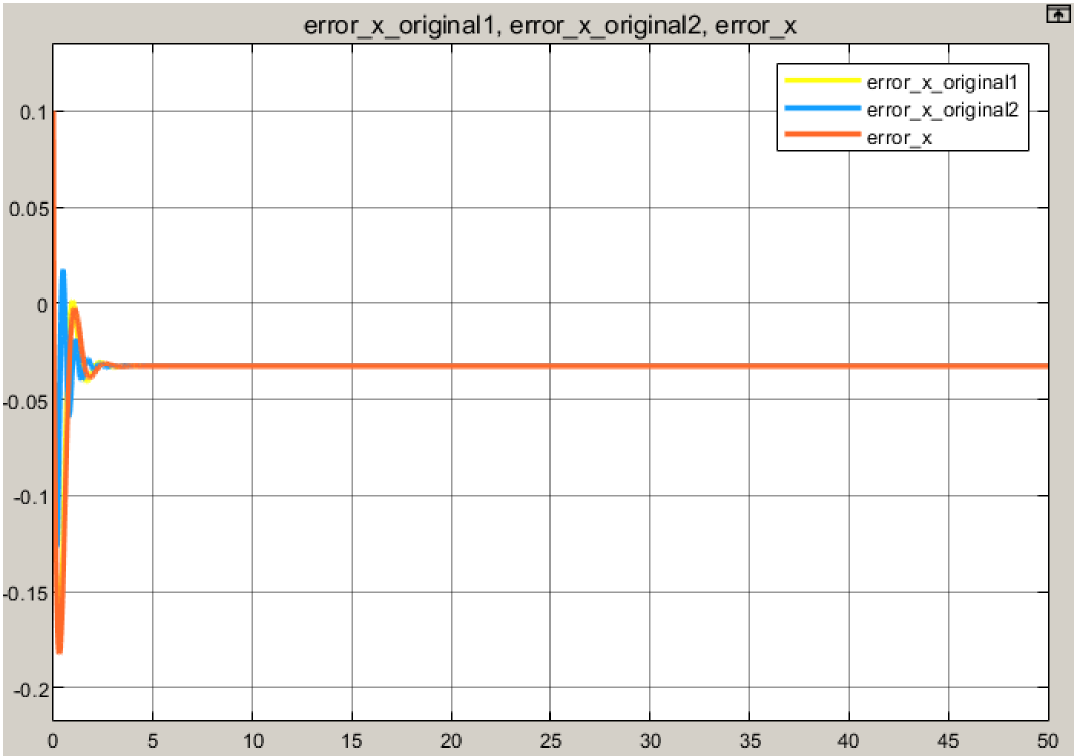
در این بخش پیش‌جبران ساز استاتیکی طراحی شده در فاز دوم را به هردو سیستم اضافه می‌کنیم.

سیستم موردنظر بدون پیش جبرانساز به خوبی و با خطای بسیار نزدیک به صفر ورودی مرجع دایره را ردیابی می‌کند که این موضوع در حالت داشتن رویتگر مرتبه کامل و مرتبه کاهش یافته نیز صدق می‌کند.

فاز چهارم:

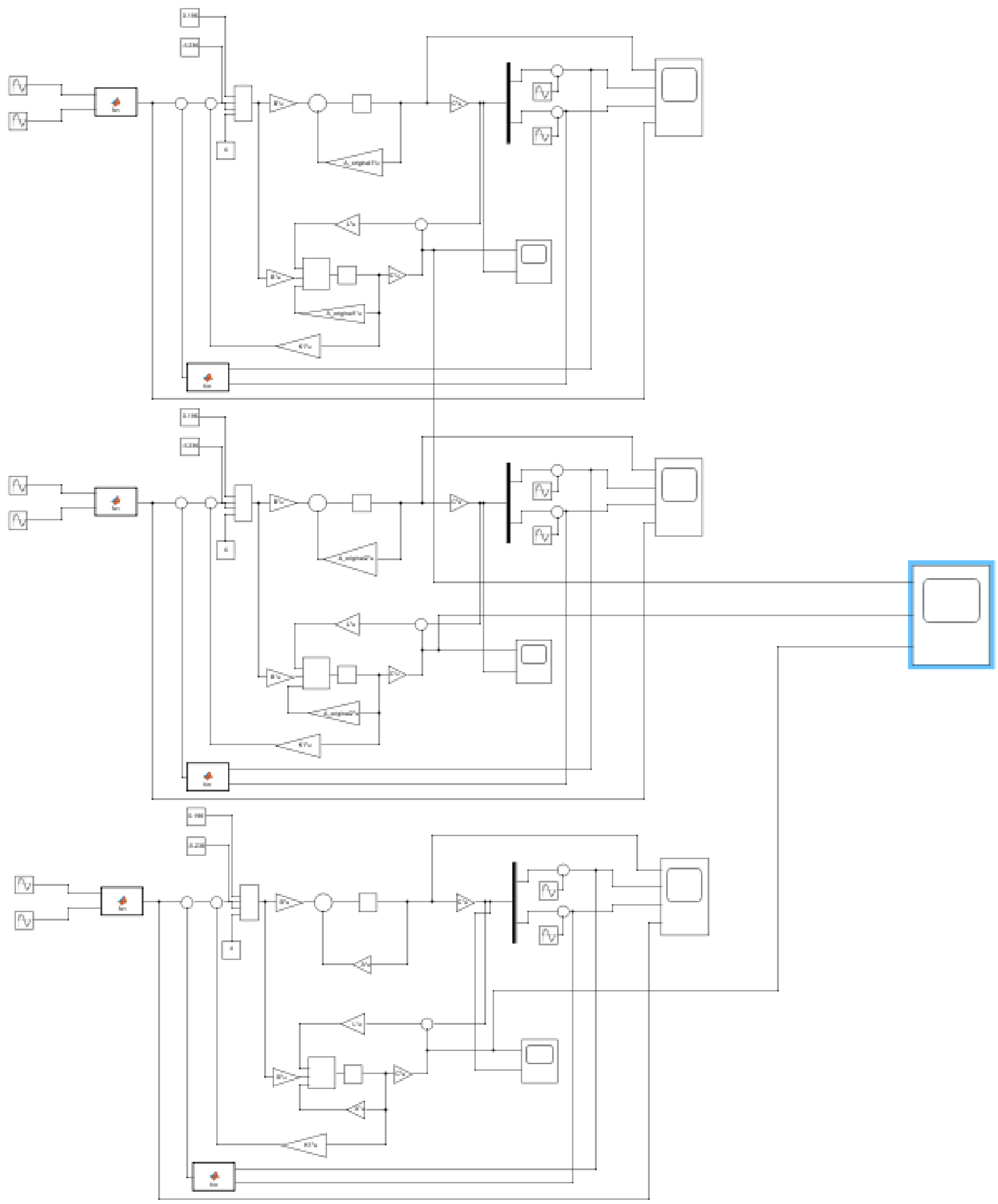
بررسی خطای تخمین حالت در صورت خطا در ماتریس حالت:

برای بررسی تغییرات خروجی در صورت وجود خطا در ماتریس حالت، ماتریس حالت را یک بار ۲۰٪ و یک بار ۴۰٪ تغییر داده‌ایم. تغییرات e_x (یکی از حالت‌های سیستم) را در سه حالت ۲۰٪ تغییر ماتریس A، ۴۰٪ تغییر ماتریس A و ماتریس بدون خطا را در زیر مشاهده می‌شود.

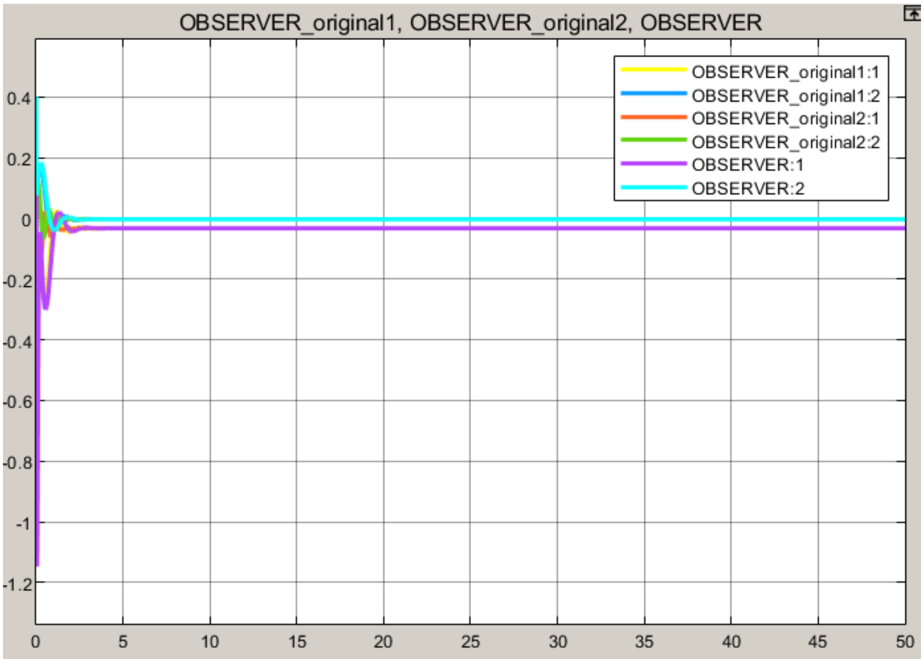


همان‌طور که مشخص است در حالت گذرای سیستم (تقریبا ۲.۵ ثانیه)، خطا در ماتریس حالت باعث خطا در حالت می‌شود ولی در حالت دائم هیچ خطایی مشاهده نمی‌شود و نمودارها بر روی هم قرار گرفته‌اند.که این به دلیل مقاومت بالای سیستم است.

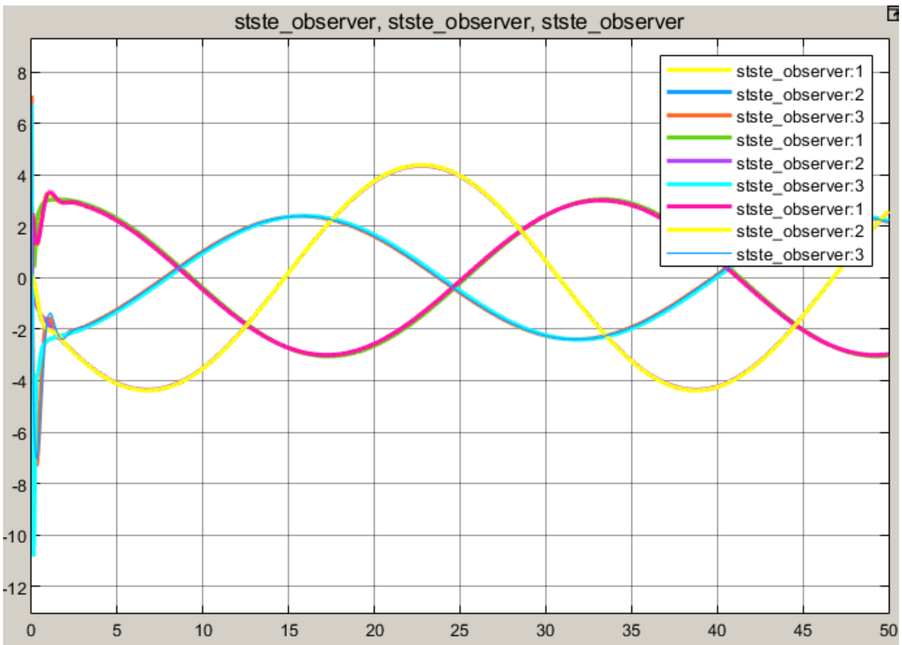
برای محاسبه خطای تخمین در صورت وجود خطا در ماتریس حالت مدار زیر در سیستم بسته شده است:



برای اینکه بتوانیم خطای تخمین را در سه حالت وجود خطا در ماتریس A محاسبه کنیم، مدار بالا بسته شد و نتیجه به صورت زیر است:



در این شکل دو حالت خروجی سیستم را که e_x و e_y می‌باشند، در سه حالت ماتریس A با تغییرات مذکور در بالا مشاهده می‌شود. با در نظر گرفتن اینکه ماتریس L در این رویتگر گین بالایی دارد، تخمین حالت‌ها به خوبی انجام شده است و این باعث شده است که حالت ها حتی با وجود تغییرات ماتریس A روی هم قرار بگیرند. حالت‌های تخمین زده شده توسط رویتگر مرتبه کاهش یافته در سه حالت ماتریس A، به صورت زیر است:



همان‌طور که در شکل بالا دیده می‌شود در حالت دائم، نمودار حالت‌ها روی هم قرار گرفته‌اند.

جمع بندی

در این پروژه، سیتم غیرخطی مورد مطالعه را خطی کرده و برای آن معادلات فضای حالت نوشتیم. سیستم پایدار مجانبی و پایدار داخلی نبود اما شرایط پایداری لیاپانوف را ارضا می‌کرد. همه حالت‌ها کنترل پذیر و رویت پذیر بودند در نتیجه سیستم مینیمال بود. برای دستیابی به اهداف کنترلی موردنظر برای سیستم فیدبک حالت طراحی کردیم. پس از ارتقا مدل سیستم دریافتیم خروجی‌ها به خوبی ورودی مرجع را دنبال می‌کنند در نتیجه نیازی به پیش جبرانساز دینامیکی برای از بین بردن خطای حالت دائم نبود. با توجه به ماتریس خروجی در نظر گرفته شده دو حالت از سه حالت سیستم در خروجی دیده می‌شد. برای تخمین حالت‌های سیستم رویتگر مرتبه کامل و مرتبه کاهش یافته طراحی کردیم. با توجه به ماتریس های به دست آمده حالت های به دست آمده توسط رویتگر تخمین مناسبی از حالت‌های سیستم بودند. سیستم در برابر خطای احتمالی ماتریس A تا حد مناسبی مقاوم بود.

مراجع

- Robust Laguerre based model predictive control of nonholonomic mobile robots under slip conditions

  By: Iman sharifi & Elnaz Firouzmand & Marzieh jamalabadi

- Laguerre based model predictive control for trajectory tracking of nonholonomic mobile robots

  By: Masoud hessamian eteffagh
- Gonzalez book