مجله پژوهش و کاربرد در مکانیک دوره ۱۳ شماره ۱ سال ۱۴۰۲ (۲۰۲۳) ۸۳-۷۱

JMRA

مجله پژوهش و کاربرد در مکانیک

کنترل جهتگیری مجری نهایی بازوی مکانیکی پایه متحرک

آرمین صفائی[1] سیدمهدی خرسندی جو[2]* احمدرضا خورشیدوند[3]

۱ فارغ التحصیل کارشناسی ارشد مهندسی مکاترونیک، واحد تهران جنوب، دانشگاه آزاد اسلامی، تهران ایران

ArminSafaei@gmail.com

۲ استادیار گروه مکانیک، واحد تهران جنوب، دانشگاه آزاد اسلامی، تهران ایران

*** نویسنده مسؤول:** *m_khorsandijou@azad.ac.ir*

۳ دانشیار گروه مکانیک، واحد تهران جنوب، دانشگاه آزاد اسلامی، تهران ایران

ar_khorshidvand@azad.ac.ir

چکیده

هدف ما در این مقاله، کنترل دوربینی است که به عنوان عملگرنهایی (End-effector) در لینک انتهایی یک بازوی رباتیکی 3R قرار دارد. پایهٔ صفر این بازوی رباتیکی بر روی سکویی با شش درجهٔ آزادی قرار دارد. سکوی شش درجه آزادی توسط یک مکانیزم شبیه سازی می‌گردد و به همراه حرکت جسمی که هدف فیلمبرداری است به صورت اغتشاش به سیستم کنترل داده می‌شود. می‌خواهیم تصویر گرفته شده توسط این دوربین، همواره در یک موقعیت مورد نظر قرار گیرد و در صورت حرکت سکوی، که در نهایت منجر به حرکت دوربین می‌شود، بازوی رباتیکی 3R با استفاده از یک کنترل کنندهٔ مناسب دوربین را به حالت اولیه برگرداند. در این مقاله ابتدا با استخراج معادلات دیفرانسیل بازوی رباتیکی 3R با پایه متحرک به روش لاگرانژین که رهیافتی بر پایه انرژی است، روش کنترل پیش بین بر حسب مدل غیر خطی (NMPC) را برای کنترل دوربین ارائه می‌کنیم.

واژه‌های کلیدی: عملگرنهایی، بازوی رباتیکی، سکوی شش درجه آزادی، کنترل پیش بین بر حسب مدل غیرخطی.

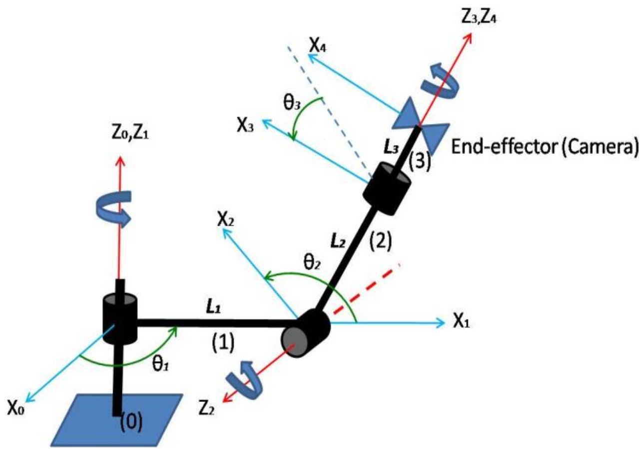
مقدمه

در سال‌های اخیر روش‌های متعددی برای کنترل بازوهای رباتیکی استفاده شده است[1]. اما در دهه گذشته Model Predictive Control به یک روش موثر در کنترل بسیاری از فرآیندها تبدیل شده است [2]. مجموعه کارهای متعددی که در [1] , [2] , [3] انجام شده نشان داده است، MPC از اقبال زیادی در حذف خطاها و اغتشاشات و همین طور عملکردهای بالا از لحاظ سرعت برخوردار است. برخلاف روش‌های کنترلی موجود که در آنها قانون کنترل بر اساس خروجی گذشته سیستم بدست می‌آید، روش کنترل پیش‌بین، یک روش بهینه مبتنی بر مدل است که از پیش‌بینی خروجی آینده سیستم برای بدست آوردن قانون کنترل استفاده می‌کند [4,5]. کنترل پیش‌بین غیرخطی، به‌دلیل این‌که از یک مدل غیرخطی برای پیش‌بینی خروجی سیستم استفاده می‌کند، فاقد پاسخ تحلیلی برای مسئله بهینه‌سازی می‌باشد و لذا باید از روش‌های بهینه‌سازی عددی برای حل مسئله بهینه‌سازی مورد استفاده قرارگیرد [4]. با این وجود، کنترل پیش‌بین غیرخطی بر خلاف کنترل پیش بین خطی، وابسته به نقطه کار سیستم نمی‌باشد؛ علاوه برآن می‌توان به‌راحتی قیود موجود در مسئله کنترلی را مد نظر گرفت.

ابتدا به بررسی ساختار هندسی و استخراج مدل دینامیکی ربات 3R با استفاد از نرم افزار Maple می‌پردازیم و سپس به حل معادلات دینامیکی غیر خطی به روش عددی با MATLAB می‌پردازیم و در ادامه به شبیه سازی سیستم حلقه باز ربات 3R پرداخته خواهد شد. در آخر با اعمال گشتاورهای کنترلی که به عنوان خروجی‌های سیستم کنترل حلقه بسته هستند و شرایط اولیه به بازوی رباتیکی، سیستم کنترل حلقه بسته را پیاده سازی می‌کنیم.

مدل‌سازی دینامیکی:

پارامترهای رابط (دناویت- هارتنبرگ) مربوط به ربات $3R$ را مطابق شکل (۱) به دست می‌آوریم.



i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	90	L_1	0	θ_2
3	90	0	L_2	θ_3
4	0	0	L_3	0

شکل (۱)-بازوی رباتیکی و جدول پارامترهای دناویت- هارتنبرگ مربوط به آن

در اینجا می‌خواهیم تبدیلی را که چهارچوب $\{i\}$ را نسبت به چهارچوب $\{i\text{-}1\}$ تعریف می‌کند، تعیین کنیم. در حالت کلی این تبدیل تابعی از چهار پارامتر رابط خواهد بود. این تبدیل به صورت رابطهٔ زیر (رابطه ۱) است:

$$ {}^{i-1}_{\;i}T = \begin{bmatrix} cos\theta_i & -sin\theta_i & 0 & a_{i-1} \\ sin\theta_i cos\alpha_{i-1} & cos\theta_i cos\alpha_{i-1} & -sin\alpha_{i-1} & -sin\alpha_{i-1} d_i \\ sin\theta_i sin\alpha_{i-1} & cos\theta_i sin\alpha_{i-1} & cos\alpha_{i-1} & cos\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (۱)$$

$$ {}^{2}_{3}T = \begin{bmatrix} cos\theta_3 & -sin\theta_3 & 0 & 0 \\ 0 & 0 & -1 & -l_2 \\ sin\theta_3 & cos\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{3}_{4}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$ {}^{0}_{1}T = \begin{bmatrix} cos\theta_1 & -sin\theta_1 & 0 & 0 \\ sin\theta_1 & cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{1}_{2}T = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 & l_1 \\ 0 & 0 & -1 & 0 \\ sin\theta_2 & cos\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

با استفاده از مقادیر پارامترهای رابط، می‌توان ماتریس‌های تبدیل کلی را برای هر رابط محاسبه کرد. سپس به وسیلهٔ ضرب ماتریس‌های تبدیل رابطها در یکدیگر، می‌توان ماتریس تبدیل کلی منفردی را که چهارچوب $\{N\}$ را به چهارچوب $\{0\}$ مربوط می‌سازد به دست آورد:

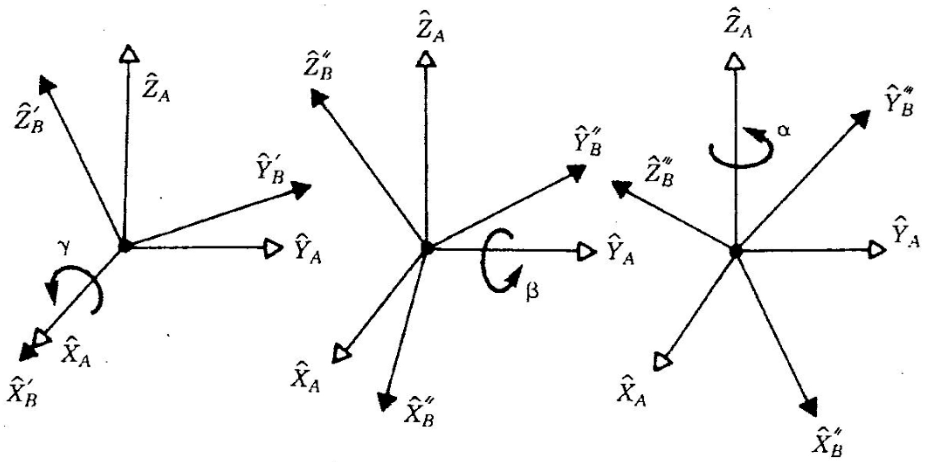
$$ {}^{0}_{N}T = {}^{0}_{1}T \; {}^{1}_{2}T \; {}^{2}_{3}T \; {}^{3}_{4}T \ldots {}^{N-1}_{\;\;N}T $$

تبدیل کلی ${}^{0}_{N}T$ در رابطهٔ فوق، تابعی از همه متغییرهای مفصلی (n متغییر) است. چنانچه مکان مفصل‌های ربات توسط حساسه‌های مکانی معین شده باشد، می‌توان مکان و جهت‌گیری آخرین رابط را با محاسبه ${}^{0}_{N}T$ مختصات دکارتی به دست آورد. حال با ضرب ماتریسی هر یک از ماتریس‌های رابطها، تبدیل کلی ${}^{0}_{4}T$ را برای ربات $3R$ محاسبه می‌کنیم.

$$ {}^{0}_{4}T = \begin{bmatrix} c\theta_1.c\theta_2.c\theta_3 + s\theta_1.s\theta_3 & -c\theta_1.c\theta_2.s\theta_3 + s\theta_1.c\theta_3 & c\theta_1.s\theta_2 & l_1.c\theta_1 + l_3.c\theta_1.s\theta_2 + l_2.c\theta_1.s\theta_2 \\ s\theta_1.c\theta_2.c\theta_3 - c\theta_1.s\theta_3 & -s\theta_1.c\theta_2.s\theta_3 - c\theta_1.c\theta_3 & s\theta_1.s\theta_2 & l_1.s\theta_1 + l_3.s\theta_1.s\theta_2 + l_2.s\theta_1.s\theta_2 \\ s\theta_2.c\theta_3 & -s\theta_2.s\theta_3 & -c\theta_2 & -l_3.c\theta_2 - l_2.c\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

روش زوایای X-Y-Z ثابت

یکی از روش‌های توصیف جهت‌گیری چهارچوب $\{B\}$ چنین است:
از چهارچوبی که برچهارچوب مرجع و معلوم $\{A\}$ منطبق است، شروع می‌کنیم. $\{B\}$ را ابتدا حول X_A به اندازه γ، سپس حول Y_A به اندازه β، و سرانجام حول Z_A به اندازه α دوران می‌دهیم.
هر یک از این سه دوران، حول محوری از چهارچوب ثابت و مرجع $\{A\}$ صورت می‌گیرد. این قرارداد مشخص کردن جهت‌گیری را، زوایای X-Y-Z ثابت می‌نامیم. کلمه ثابت، به ثابت بودن چهارچوبی که دوران‌ها حول محورهای آن انجام می‌گیرند، اشاره می‌کند شکل (۲). گاهی از این قرارداد با نام قرارداد « زوایای رول[۱] ، پیچ[۲] ، و یاو[۳] » یاد می‌شود. اما این نام در قراردادهای دیگر هم کاربرد دارد.



شکل(۲) زوایای رول، پیچ، یاو حول محورهای ثابت

به دست آوردن ماتریس دوران معادل ${}^{A}_{B}P_{XYZ}(\gamma,\beta,\alpha)$ به سادگی صورت می‌گیرد، زیرا دوران‌ها همگی حول محورهای چهارچوب مرجع انجام می‌شوند.

$${}^{A}_{B}R_{XYZ}(\gamma,\beta,\alpha)=R_Z(\alpha)\,R_Y(\beta)\,R_X(\gamma)$$

$$=\begin{bmatrix} Cos\alpha & -Sin\alpha & 0\\ Sin\alpha & Cos\alpha & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix} Cos\beta & 0 & Sin\beta\\ 0 & 1 & 0\\ -Sin\beta & 0 & Cos\beta\end{bmatrix}\begin{bmatrix} 1 & 0 & 0\\ 0 & Cos\gamma & -Sin\gamma\\ 0 & Sin\gamma & Cos\gamma\end{bmatrix}$$

با ضرب کردن ماتریس‌ها نتیجه می‌شود:

$${}^{A}_{B}R_{XYZ}(\gamma,\beta,\alpha)=\begin{bmatrix} C\alpha C\beta & C\alpha S\beta S\gamma-S\alpha C\gamma & C\alpha S\beta C\gamma+S\alpha S\gamma\\ S\alpha C\beta & S\alpha S\beta S\gamma+C\alpha C\gamma & S\alpha S\beta C\gamma-C\alpha S\gamma\\ -S\beta & C\beta S\gamma & C\beta C\gamma\end{bmatrix}$$

که در آن $C\alpha$ مخفف $Cos\alpha$، و $S\alpha$ مخفف $Sin\alpha$ و... است.

بیان ماتریس تبدیل کلی ${}^{E}_{4}T$ با استفاده از روش زوایای X-Y-Z ثابت

---

۱ . roll دوران حول محور طولی
۲ . pitch دوران حول محور عرضی
۳ . yaw دوران حول محور قائم

ماتریس ${}^{0}_{4}T$ ، بیانگر تبدیل کلی از عملگر نهایی (که همان دوربین باشد) تا پایه ثابت ربات (پایه صفر) است. اما با توجه به متحرک بودن پایهٔ صفر ربات نسبت به زمین، تبدیل دیگری از پایه‌ی صفر تا زمین(E) داریم که آن را با ${}^{E}_{0}T$ نشان می‌دهیم.

با توجه به توصیف جهت‌گیری یک چهارچوب نسبت به چهارچوب دیگر در روش زوایای X-Y-Z، ماتریس ${}^{E}_{0}T$ را به صورت زیر تعریف می‌کنیم:

$$ {}^{E}_{0}T = \left[\begin{array}{ccc|c} c\alpha.c\beta & c\alpha.s\beta.s\gamma - s\alpha.c\gamma & c\alpha.s\beta.c\gamma + s\alpha.s\gamma & X_0 \\ s\alpha.c\beta & s\alpha.s\beta.s\gamma + c\alpha.c\gamma & s\alpha.s\beta.c\gamma - c\alpha.s\gamma & Y_0 \\ -s\beta & c\beta.s\gamma & c\beta.c\gamma & Z_0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

همان‌طور که می‌بینیم ماتریس ${}^{E}_{0}T$، شامل یک دوران و یک انتقال است شکل (۴-۸). در نتیجه ماتریس تبدیل کلی از عملگر نهایی $\{4\}$ تا زمین (E)، یعنی ${}^{E}_{4}T$ به صورت زیر خواهد بود:

$$ {}^{E}_{0}T\ {}^{0}_{4}T = {}^{E}_{4}T $$

$$ {}^{E}_{4}T = \begin{bmatrix} c\alpha.c\beta & c\alpha.s\beta.s\gamma - s\alpha.c\gamma & c\alpha.s\beta.c\gamma + s\alpha.s\gamma & X_0 \\ s\alpha.c\beta & s\alpha.s\beta.s\gamma + c\alpha.c\gamma & s\alpha.s\beta.c\gamma - c\alpha.s\gamma & Y_0 \\ -s\beta & c\beta.s\gamma & c\beta.c\gamma & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ \begin{bmatrix} c\theta_1.c\theta_2.c\theta_3 + s\theta_1.s\theta_3 & -c\theta_1.c\theta_2.s\theta_3 + s\theta_1.c\theta_3 & c\theta_1.s\theta_2 & l_1.c\theta_1 + l_3.c\theta_1.s\theta_2 + l_2.c\theta_1.s\theta_2 \\ s\theta_1.c\theta_2.c\theta_3 - c\theta_1.s\theta_3 & -s\theta_1.c\theta_2.s\theta_3 - c\theta_1.c\theta_3 & s\theta_1.s\theta_2 & l_1.s\theta_1 + l_3.s\theta_1.s\theta_2 + l_2.s\theta_1.s\theta_2 \\ s\theta_2.c\theta_3 & -s\theta_2.s\theta_3 & -c\theta_2 & -l_3.c\theta_2 - l_2.c\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

رابطهٔ سرعت‌ها را می‌توان نسبت به چهارچوب $\{i\}$ چنین نوشت:

$$ {}^{i}\omega_{i+1} = {}^{i}\omega_i + {}^{i}_{i+1}R\ \dot{\theta}_{i+1}\ {}^{i+1}Z_{i+1} \qquad (۲) $$

$$ \dot{\theta}_{i+1}\ {}^{i+1}Z_{i+1} = {}^{i+1}\begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix} \qquad (۳) $$

از ماتریس دوران ارتباط دهندهٔ چهارچوب‌های i و $i+1$ ، برای نشان دادن مولفهٔ دورانی اضافه شده بر اثر حرکت مفصل در چهارچوب $\{i\}$، استفاده کرده‌ایم. ماتریس دوران، محور دوران مفصل $i+1$ در چهارچوب $\{i\}$ توصیف می‌کند. بنابراین می‌توان دو مولفه‌ی سرعت زاویه‌ای را با هم جمع کرد. با پیش ضرب هر دو طرف معادلهٔ (۵-۱) در ${}^{i+1}_{i}R$، می‌توان سرعت زاویه‌ای رابط $i+1$ را در چهارچوب $\{i+1\}$ چنین توصیف کرد:

$$ {}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\ {}^{i}\omega_i + \dot{\theta}_{i+1}\ {}^{i+1}Z_{i+1} \qquad (۴) $$

سرعت خطی مبدأ چهارچوب $\{i+1\}$ برابر است با سرعت خطی مبدأ چهارچوب $\{i\}$ به علاوه مولفهٔ جدیدی که بر اثر سرعت دورانی رابط i حاصل می‌شود. پس داریم:

$$ {}^{i}v_{i+1} = {}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1} \qquad (۵) $$

با ضرب هر دو طرف در ${}^{i+1}_{i}R$ نتیجه می‌شود:

$$ {}^{i+1}v_{i+1} = {}^{i+1}_{i}R(\ {}^{i}v_i + {}^{i}\omega_i \times {}^{i}P_{i+1}) \qquad (۶) $$

با رجوع به قسمت مربوط به انتقال و دوران پایهٔ بازو رباتیکی (0) نسبت به زمین (E) ، سرعت زاویه‌ای پایهٔ بازو به صورت زیر تعریف می‌شود:

$$^{0}\omega_0 = \dot{\gamma}\begin{bmatrix}1\\0\\0\end{bmatrix} + \dot{\beta}\begin{bmatrix}1 & 0 & 0\\0 & C\gamma & S\gamma\\0 & -S\gamma & C\gamma\end{bmatrix}\begin{bmatrix}0\\1\\0\end{bmatrix} + \dot{\alpha}\begin{bmatrix}1 & 0 & 0\\0 & C\gamma & S\gamma\\0 & -S\gamma & C\gamma\end{bmatrix}\begin{bmatrix}C\beta & 0 & -S\beta\\0 & 1 & 0\\S\beta & 0 & C\beta\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}$$

با توجه به رابطهٔ (۴) و ماتریس‌های تبدیل به دست آمده، سرعت‌های زاویه‌ای هر رابط به قرار زیر است:

$$^{1}_{0}R = \begin{bmatrix}C\theta_1 & S\theta_1 & 0\\-S\theta_1 & C\theta_1 & 0\\0 & 0 & 1\end{bmatrix} \quad , \quad ^{1}\omega_1 = {}^{1}_{0}R\ {}^{0}\omega_0 + \begin{bmatrix}0\\0\\\dot{\theta}_1\end{bmatrix}$$

$$^{2}_{1}R = \begin{bmatrix}C\theta_2 & 0 & S\theta_2\\-S\theta_2 & 0 & C\theta_2\\0 & -1 & 0\end{bmatrix} \quad , \quad ^{2}\omega_2 = {}^{2}_{1}R\ {}^{1}\omega_1 + \begin{bmatrix}0\\0\\\dot{\theta}_2\end{bmatrix}$$

$$^{3}_{2}R = \begin{bmatrix}C\theta_3 & 0 & S\theta_3\\-S\theta_3 & 0 & C\theta_3\\0 & -1 & 0\end{bmatrix} \quad , \quad ^{3}\omega_3 = {}^{3}_{2}R\ {}^{2}\omega_2 + \begin{bmatrix}0\\0\\\dot{\theta}_3\end{bmatrix}$$

$$^{4}_{3}R = \begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix} \quad , \quad ^{4}\omega_4 = {}^{4}_{3}R\ {}^{3}\omega_3 + \begin{bmatrix}0\\0\\0\end{bmatrix}$$

با توجه به رابطهٔ (۶) سرعت خطی هر رابط نیز، از رابطه‌های زیر حساب می‌شود:

$$^{0}v_0 = \begin{bmatrix}\dot{x}\\\dot{y}\\\dot{z}\end{bmatrix} \quad , \quad ^{0}P_1 = \begin{bmatrix}0\\0\\0\end{bmatrix} \quad , \quad ^{1}v_1 = {}^{1}_{0}R(\,^{0}v_0 + {}^{0}\omega_0 \times {}^{0}P_1)$$

$$^{1}P_2 = \begin{bmatrix}L_1\\0\\0\end{bmatrix} \quad , \quad ^{2}v_2 = {}^{2}_{1}R(\,^{1}v_1 + {}^{1}\omega_1 \times {}^{1}P_2)$$

$$^{2}P_3 = \begin{bmatrix}0\\-L_2\\0\end{bmatrix} \quad , \quad ^{3}v_3 = {}^{3}_{2}R(\,^{2}v_2 + {}^{2}\omega_2 \times {}^{2}P_3)$$

$$^{3}P_4 = \begin{bmatrix}0\\0\\L_3\end{bmatrix} \quad , \quad ^{4}v_4 = {}^{4}_{3}R(\,^{3}v_3 + {}^{3}\omega_3 \times {}^{3}P_4)$$

توجه داشته باشید، تمامی مقادیر به دست آمده از $^{i+1}\omega_{i+1}$ و $^{i+1}v_{i+1}$ که در بالا ذکر شده‌اند، در محاسبهٔ $^{i}v_{i+1}$ ، مورد استفاده قرار می‌گیرند. داریم:

$$^{0}v_1 = {}^{0}v_0 + {}^{0}\omega_0 \times {}^{0}P_1$$

$$^{1}v_2 = {}^{1}v_1 + {}^{1}\omega_1 \times {}^{1}P_2$$

$$^{2}v_3 = {}^{2}v_2 + {}^{2}\omega_2 \times {}^{2}P_3$$

$$^{3}v_4 = {}^{3}v_3 + {}^{3}\omega_3 \times {}^{3}P_4$$

۳-۵- روش لاگرانژ در به دست آوردن معادله‌های دینامیکی بازوی مکانیکی

در این بخش، به طور خلاصه روشی را برای به دست آوردن معادله‌های دینامیکی، به نام فرمولبندی دینامیکی لاگرانژ، معرفی خواهیم کرد. روش لاگرانژ رهیافتی بر پایهٔ انرژی دارد. بررسی ما از دینامیک لاگرانژی مختصر و مربوط به حالت خاص یک بازوی مکانیکی زنجیری با رابطه‌های صلب خواهد بود [۳] .

ابتدا عبارتی برای انرژی جنبشی بازوی مکانیکی به دست می‌آوریم. انرژی جنبشی رابط iام، k_i را می‌توان چنین نوشت:

$$k_i = \frac{1}{2} m_i V_{C_i}^T V_{C_i} + \frac{1}{2}\ {}^i\omega_i^T\ {}^{C_i}I_i\ {}^i\omega_i \qquad (۷)$$

که در آن جملهٔ اول، انرژی جنبشی حاصل از سرعت خطی مرکز جرم رابط، و جملهٔ دوم، انرژی جنبشی ناشی از سرعت دورانی رابط است. انرژی جنبشی کل بازو برابر با مجموع انرژی‌های جنبشی هر یک از رابطه‌هاست. یعنی:

$$k_i = \sum_{i=0}^{n} \frac{1}{2} m_i V_{C_i}^T V_{C_i} + \frac{1}{2}\ {}^i\omega_i^T\ {}^{C_i}I_i\ {}^i\omega_i$$

چون جمله‌های V_{C_i} و ${}^i\omega_i$ در (۵-۶) توابعی از Θ و $\dot{\Theta}$ هستند، انرژی جنبشی هر بازوی مکانیکی ماهر را می‌توان با عبارتی اسکالر به صورت تابعی از مکان و سرعت مفصل‌ها، $k(\Theta,\dot{\Theta})$، بیان کرد. بدین ترتیب انرژی جنبشی بازو را می‌توان چنین نوشت:

$$k(\Theta,\Theta^{\cdot}) = \frac{1}{2} \Theta^{\cdot T} M(\Theta) \Theta^{\cdot} \qquad (۸)$$

انرژی پتانسیل رابط iام، u_i را می‌توان چنین نوشت:

$$u_i = -m_i\ {}^0g^T\ {}^0P_{C_i}$$

که در آن ${}^0g^T$ بردار ۳×۱ گرانی، و ${}^0P_{C_i}$ بردار نشان دهندهٔ مکان مرکز جرم رابط iام، (که ما مرکز جرم هر رابط را در ربات، در ابتدای آن رابط در نظر گرفته‌ایم) است. انرژی پتانسیل کل ذخیره شده در بازو برابر با مجموع انرژی‌های پتانسیل هر یک از رابطه‌هاست:

$$u_g = \sum_{i=0}^{n} (-m_i\ {}^0g^T\ {}^0P_{C_i}) \qquad (۹)$$

با فرض این که اگر در مفصل‌های ربات از فنر استفاده شود، رابطهٔ انرژی پتانسیل کشسانی به صورت زیر خواهد بود:

$$u_e = \frac{1}{2} \sum_{i=1}^{n} K_i \theta_i^2 \qquad (۱۰)$$

که در آن K_i، ثابت فنر مفصل iام است. پس در مجموع انرژی پتانسیل به صورت زیر است:

$$u = u_g + u_e$$

روش دینامیکی لاگرانژ [۳] ، ابزاری را برای به دست آوردن معادله‌های حرکت از تابعی اسکالر به نام لاگرانژین، که به صورت اختلاف بین انرژی‌های پتانسیل و جنبشی یک سیستم مکانیکی تعریف شده است، فراهم می‌آورد. مطابق نمادگذاری ما، لاگرانژین هر بازوی مکانیکی چنین خواهد بود:

$$\mathcal{L}(\Theta,\dot{\Theta}) = k(\Theta,\dot{\Theta}) - u(\Theta) \qquad (۱۱)$$

پس معادله‌های حرکت بازو از رابطهٔ زیر به دست می‌آیند:

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\Theta}}-\frac{\partial \mathcal{L}}{\partial \Theta}=\tau$$

بعد از به دست آوردن معادلات حرکت بازوی رباتیکی $3R$، باید به حل معادلات با استفاده از MATLAB بپردازیم. فرم کلی معادلات را به صورت زیر بیان می‌کنیم:

$$[A(\Theta)]_{9\times9}[\ddot{\Theta}]_{9\times1}=[-B(\Theta,\dot{\Theta},\tau)]_{9\times1} \qquad (۱۲)$$

که در آن ماتریس A شامل فقط متغییرهای مکان $\Theta=(x\,,y\,,z\,,\alpha\,,\beta\,,\gamma\,,\theta_1\,,\theta_2\,,\theta_3)$، ماتریس B شامل متغییرهای مکان، سرعت‌ها $\dot{\Theta}=(\dot{x}\,,\dot{y}\,,\dot{z}\,,\dot{\alpha}\,,\dot{\beta}\,,\dot{\gamma}\,,\dot{\theta}_1\,,\dot{\theta}_2\,,\dot{\theta}_3)$ و گشتاورهای سیستم (τ_1,τ_2,τ_3)، و ماتریس $\ddot{\Theta}=\begin{bmatrix}\ddot{X} & \ddot{Y} & \ddot{Z} & \ddot{\alpha} & \ddot{\beta} & \ddot{\gamma} & \ddot{\theta}_1 & \ddot{\theta}_2 & \ddot{\theta}_3\end{bmatrix}^T$ ماتریس شتاب‌های سیستم است.

در این معادله، A یک ماتریس 9×9 است که سطرهای آن به ترتیب شامل ضرایب $\ddot{X}$، $\ddot{Y}$، $\ddot{Z}$، $\ddot{\alpha}$، $\ddot{\beta}$، $\ddot{\gamma}$، $\ddot{\theta}_1$، $\ddot{\theta}_2$، و $\ddot{\theta}_3$ معادلات است. برای مثال درآیهٔ A_{11}، ضریب $\ddot{X}$ در معادلهٔ اول، و A_{12}، ضریب $\ddot{Y}$ در معادلهٔ اول، و به همین ترتیب A_{19}، ضریب $\ddot{\theta}_3$ در معادله‌ی اول خواهد بود:

$$\begin{bmatrix}
A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} & A_{17} & A_{18} & A_{19}\\
A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} & A_{27} & A_{28} & A_{29}\\
A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & A_{36} & A_{37} & A_{38} & A_{39}\\
A_{41} & A_{42} & A_{43} & A_{44} & A_{45} & A_{46} & A_{47} & A_{48} & A_{49}\\
A_{51} & A_{52} & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} & A_{58} & A_{59}\\
A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} & A_{67} & A_{68} & A_{69}\\
A_{71} & A_{72} & A_{73} & A_{74} & A_{75} & A_{76} & A_{77} & A_{78} & A_{79}\\
A_{81} & A_{82} & A_{83} & A_{84} & A_{85} & A_{86} & A_{87} & A_{88} & A_{89}\\
A_{91} & A_{92} & A_{93} & A_{94} & A_{95} & A_{96} & A_{97} & A_{98} & A_{99}
\end{bmatrix}
\begin{bmatrix}\ddot{X}\\ \ddot{Y}\\ \ddot{Z}\\ \ddot{\alpha}\\ \ddot{\beta}\\ \ddot{\gamma}\\ \ddot{\theta}_1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix}
=
\begin{bmatrix}B_{11}\\ B_{21}\\ B_{31}\\ B_{41}\\ B_{51}\\ B_{61}\\ B_{71}\\ B_{81}\\ B_{91}\end{bmatrix}$$

ماتریس B، یک ماتریس 9×1 است که هر سطر آن از رابطهٔ زیر محاسبه می‌شود:

$$B(i,1)=\Big(Left\ hand\ side\ Equation(i)-\ddot{X}A(i,1)-\ddot{y}A(i,2)-\ddot{z}A(i,3)-\ddot{\alpha}A(i,4)-\ddot{\beta}A(i,5)-\ddot{\gamma}A(i,6)-\ddot{\theta}_1A(i,7)$$
$$-\ddot{\theta}_2A(i,8)-\ddot{\theta}_3A(i,9)\Big) \quad , \quad i=(1\dots9)$$

حل معادلهٔ دیفرانسیل بازوی رباتیکی به روش عددی در MATLAB

هدف ما در کنترل بازوی رباتیکی 3R این است که دوربین متصل به عملگرنهایی، همیشه هدف مورد نظر را در مرکز قاب خود داشته باشد. پس طراحی کنترل کنندهٔ ما هم بر مبنای این هدف صورت می‌گیرد.

برای این منظور باید موقعیت هدف، نسبت به عملگرنهایی (دوربین) را داشته باشیم. با توجه به ماتریس‌های تبدیل گفته شده در رابطه ۱، موقعیت هدف نسبت به عملگرنهایی $\{4\}$ را، با ماتریس تبدیل ${}_{object}^{4}T$ بیان می‌کنیم. حال این سوال مطرح است که ${}_{object}^{4}T$ را چگونه محاسبه کنیم؟ برای پاسخ به این پرسش از رابطهٔ زیر استفاده می‌کنیم:

$$[{}_{object}^{4}T]_{4\times4}=[{}_{E}^{4}T]_{4\times4}\,[{}_{object}^{E}T]_{4\times4}$$

با توجه به روش گفته شده در بخش (۴-۶) در وارون کردن ماتریس‌های تبدیل، ماتریس تبدیل ${}_{E}^{4}T$ را از وارون کردن ${}_{4}^{E}T$ حساب می‌کنیم.

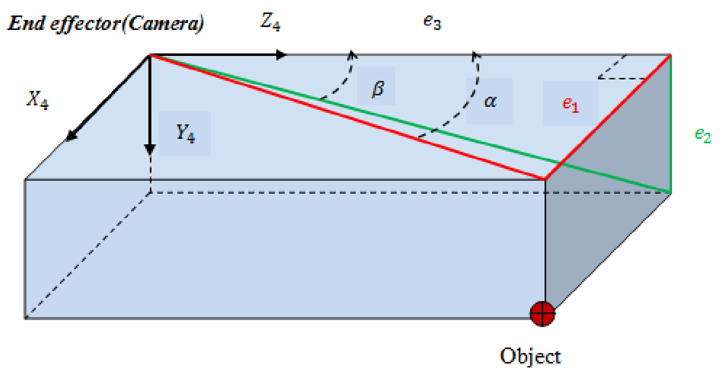
بعد از محاسبه ماتریس ${}_{E}^{4}T$، باید ماتریس ${}_{object}^{E}T$ را که بیانگر موقعیت هدف نسبت به زمین است تعریف کنیم. این ماتریس را به صورت زیر بیان می‌کنیم:

$$ {}_{object}^{E}T = \left[\begin{array}{ccc|c} 1 & 0 & 0 & d_1 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 1 & d_3 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

این ماتریس به این معنی است که موقعیت هدف نسبت به زمین، فقط دارای یک انتقال است. همانطور که می‌دانیم d_1، d_2 و d_3 بیانگر مختصات هدف نسبت به زمین هستند. با ضرب دو ماتریس ${}_{E}^{4}T$ و ${}_{object}^{E}T$ در یکدیگر، ماتریس تبدیل ${}_{object}^{4}T$ حاصل می‌شود.

حال که به موقعیت **هدف نسبت به دوربین** (${}_{object}^{4}T$) رسیدیم، با توجه به شکل بازوی رباتیکی و با فرض آنکه محور Z_4 از مرکز لنز دوربین می‌گذرد، کافیست محور Z عملگر نهایی (یعنی همان محور Z_4 دوربین) بر محور Z هدف منطبق شود. در این صورت است که هدف همواره در مرکز قاب دوربین قرار می‌گیرد. برای این منظور باید زوایای α و β صفر شوند تا محور Z_4 بر روی هدف منطبق شود شکل (۳).

در شکل زیر فرض بر این است که عملگرنهایی و هدف در موقعیت‌های نشان داده شده قرار دارند.



شکل (۳) زوایایی که باید در کنترل بازو صفر شوند، α و β هستند.

با توجه به ماتریس ${}_{object}^{4}T$ به دست آمده در صفحهٔ قبل، ${}_{object}^{4}T(1,4)$ ، ${}_{object}^{4}T(2,4)$ و ${}_{object}^{4}T(3,4)$ را به ترتیب e_1 ، e_2 و e_3 (همانطور که قبلاً گفته شد این پارامترها بیانگر فاصلهٔ هدف در مختصات دوربین هستند) و مشتقات این سه مؤلفه را هم e_{1d} ، e_{2d} و e_{3d} می‌نامیم.

برای صفر شدن زوایای α و β ، باید تانژانت این دو زاویه صفر شود تا محور Z_4 دوربین بر روی هدف قرار گیرد. با توجه به شکل (۳) داریم:

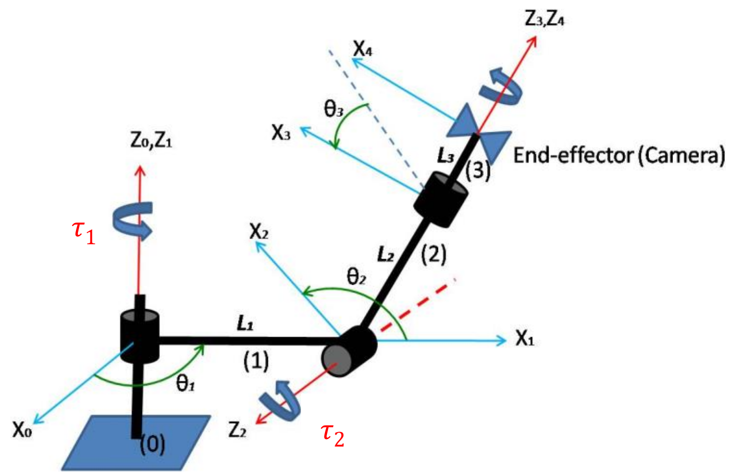
$$ tan\, \alpha = \frac{e_1}{e_3} = 0 $$

$$ tan\beta = \frac{e_2}{e_3} = 0 $$

$tan\, \alpha$ و $tan\, \beta$ را به ترتیب $error_1$ و $error_2$ نامگذاری می‌کنیم. حال اگر $error_1$ و $error_2$ را بتوانیم صفر کنیم عمل کنترل صورت گرفته است. برای این منظور باید محورهای X_4 و Y_4 ربات را طوری حرکت دهیم تا محور Z_4 بر روی هدف قرار گیرد. مطابق شکل (۴) و با این فرض که کاراندازهای[۱] بازوی رباتیکی در مفصل‌های آن تعبیه شده است، این عمل به ترتیب توسط گشتاورهای τ_2 و τ_1 صورت می‌گیرد. (جهت محور Y_4 به سمت بیرون صفحه است).

توجه به این نکته مهم است که بدون پی بردن به این موضوع، که برای صفر کردن هر یک از خطاها چه گشتاورهایی لازم است، کنترل بازوی رباتیکی برای رسیدن به موقعیت مطلوب امکان پذیر نخواهد بود.

1. Actuator



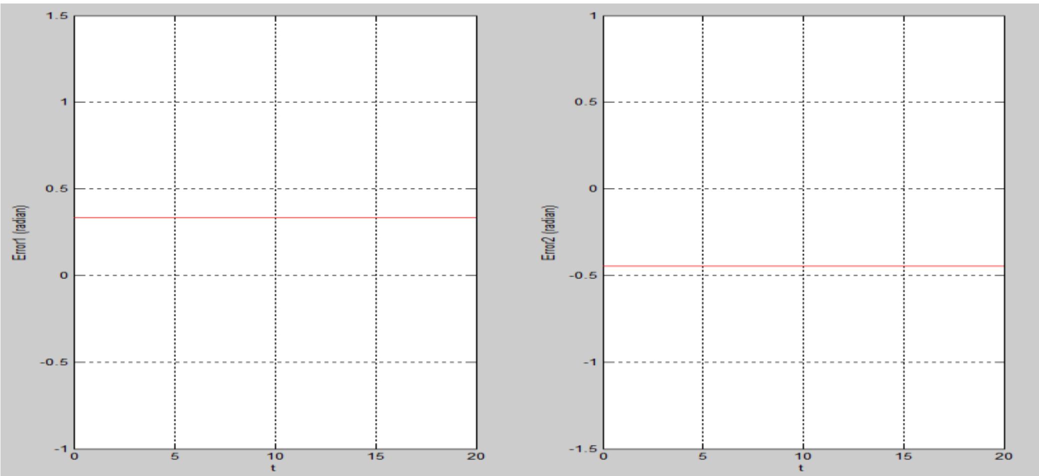
شکل (۴) گشتاورهای τ_1 و τ_2 به ترتیب در صفر کردن خطاهای $error_2$ و $error_1$.

پیاده سازی سیستم حلقه باز:

جدول متغییرهای مربوط به مثال اول:

Mass(kg)				Lengths(meter)			Spring K			Damping			g	J	Object distance		
M_0	M_1	M_2	M_3	L_1	L_2	L_3	K_1	K_2	K_3	C_1	C_2	C_3	g	J	d_1	d_2	$\boldsymbol{d_3}$
20	1	1	1	0.25	0.25	0.25	0	0	0	0	0	0	0	2	300	400	-900

	Positions									Velocities								
State variable	x	y	z	α	β	γ	θ_1	θ_2	θ_3	$\dot{x}$	$\dot{y}$	$\dot{z}$	$\dot{\alpha}$	$\dot{\beta}$	$\dot{\gamma}$	$\dot{\theta}_1$	$\dot{\theta}_2$	$\boldsymbol{\dot{\theta}_3}$
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

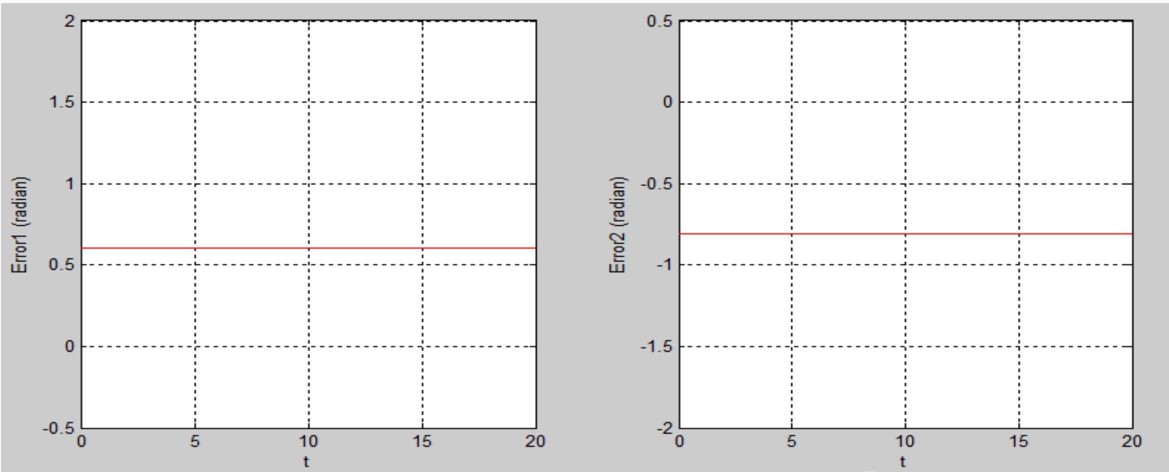


شکل (۵)- مشاهده خطا ها ($error_2$ و $error_1$) در سیستم حلقه باز

جدول متغییرهای مربوط به مثال دوم:

Mass(kg)				Lengths(meter)			Spring K			Damping			g	J	Object distance		
M_0	M_1	M_2	M_3	L_1	L_2	L_3	K_1	K_2	K_3	C_1	C_2	C_3	g	J	d_1	d_2	d_3
20	1	1	1	0.25	0.25	0.25	0	0	0	0	0	0	0	2	300	400	5

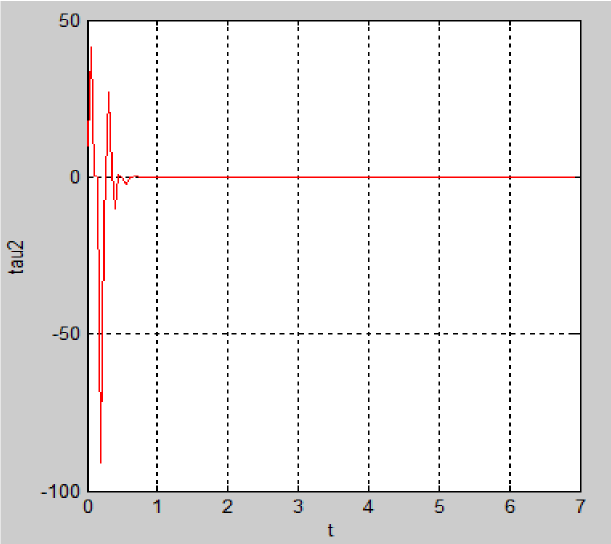
	Positions									Velocities								
State variable	x	y	z	α	β	γ	θ_1	θ_2	θ_3	$\dot{x}$	$\dot{y}$	$\dot{z}$	$\dot{\alpha}$	$\dot{\beta}$	$\dot{\gamma}$	$\dot{\theta}_1$	$\dot{\theta}_2$	$\dot{\theta}_3$
	0	0	500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

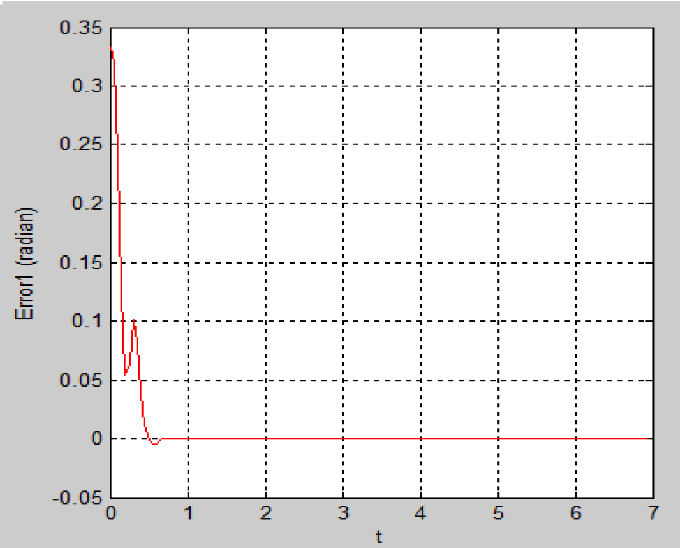


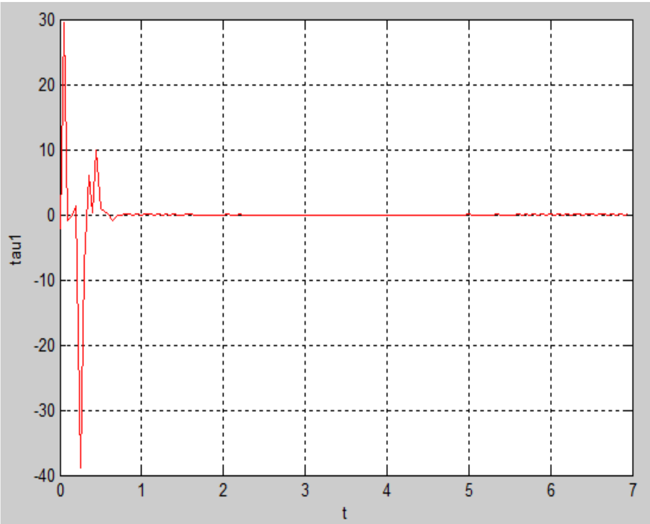
شکل (۶)– مشاهده خطا ها ($error_2$ و $error_1$) در سیستم

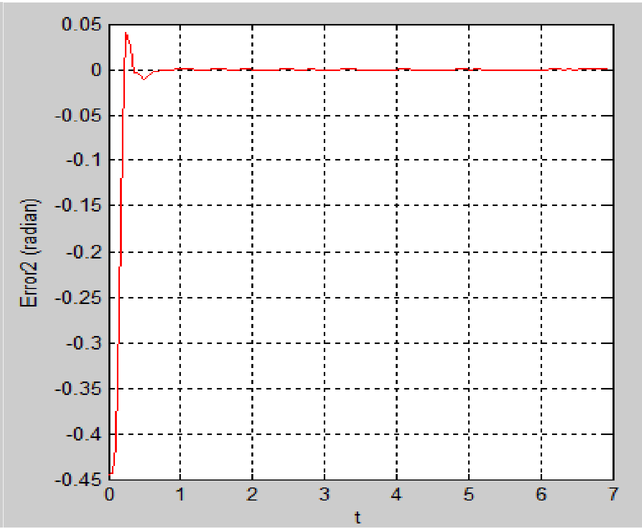
پیاده سازی سیستم حلقه بسته:

با توجه به پارامترهای جدول مثال ۱ گراف‌های خطا و گشتاورهای اعمال شده برای صفر کردن خطاها به صورت زیر هستند:

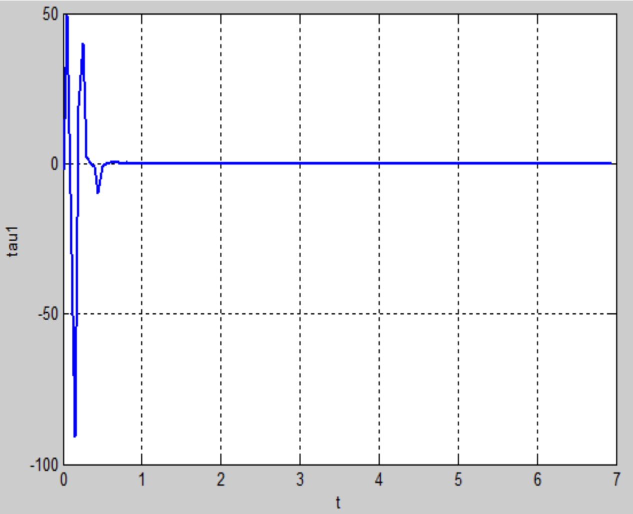

tau2
t

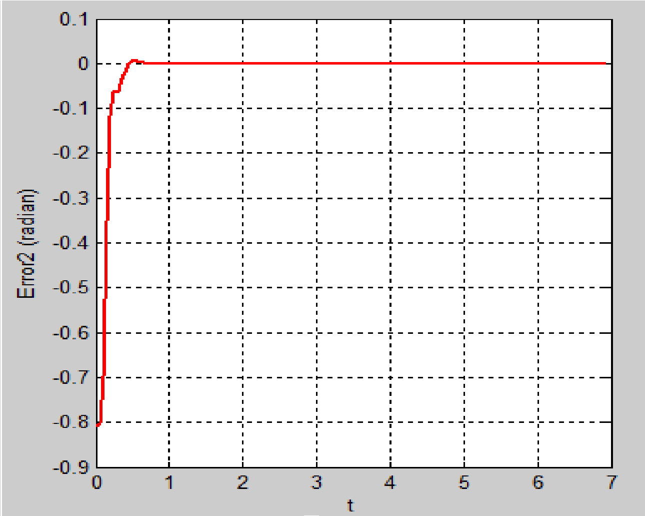

Error1 (radian)
t

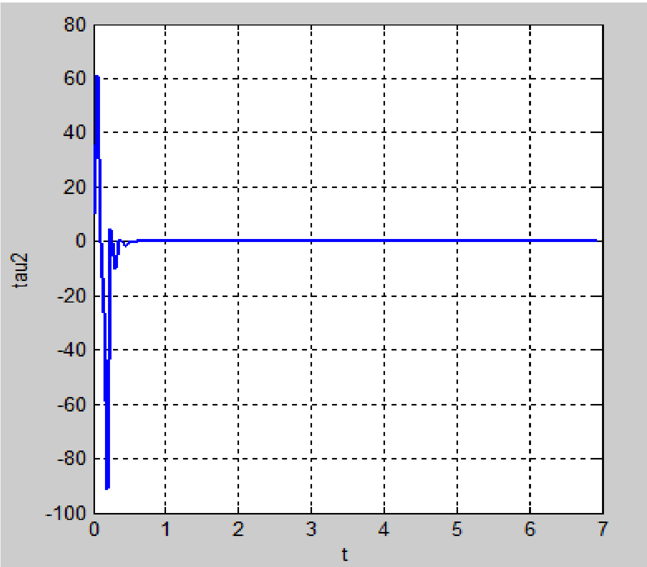

tau1
t

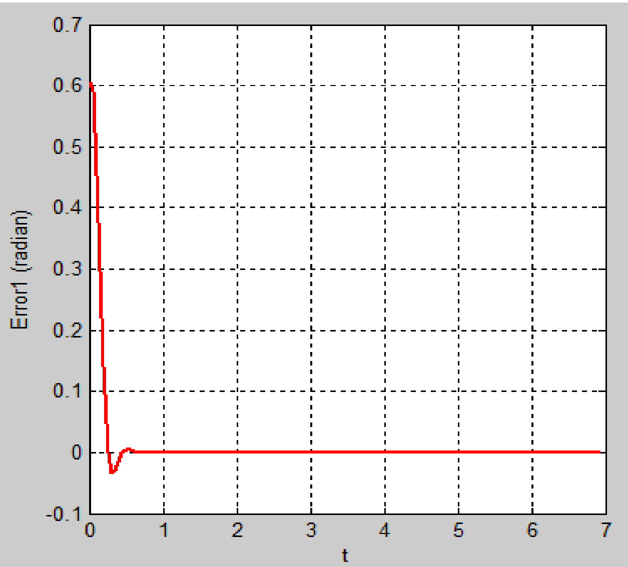

Error2 (radian)
t

با توجه به پارامترهای جدول مثال ۲ گراف‌های خطا و گشتاورهای اعمال شده برای صفر کردن خطاها به صورت زیر هستند:









[1] CANUDAS de WIT C., SICILANO B., BASTIN G., *Theory of robot Control*, Springer Verlag, 1996.

[2] ALLGOWER F., BADGWELL T.A., QIN J.S., RAWLINGS J.B., WRIGHT S.J., *Nonlinear Predictive Control and Moving Horizon Estimation – An Introduction Overview*, ECC99, pp. 391-449.

[۳] W. Mayer and P. Benedict, "Path Planning and the Geometry of Joint Space Obstacles", *IEEE International Conference on Robotics and Automation,* pp.215- 219, 1988.

[4] C. Warren, A Vector Based Approach to Robot Path Planning", *IEEE International Conference on Robotics and Automation*, pp.1021-1026, 1991.

[5] T. L. Perez and M.A. Wesley, "An Algorithm for Planning Collision Free Paths Among Polyhedral Obstacles", *Communications of the ACM,* Vol.22, no.10, pp.219-291. Sept, 1992.