مجله مهندسی مکانیک مدرس، اردیبهشت 1394، دوره 15، شماره 2، ص‌ص 147-158

ماهنامه علمی پژوهشی

مهندسی مکانیک مدرس

mme.modares.ac.ir

کنترل ربات موازی 3-RPR با استفاده از روش مود لغزشی بر روی مسیر بهینه‌سازی شده توسط الگوریتم بهینه سازی فاخته

سید علیرضا معزی[1]، منصور رفیعیان[2*]، سعید ابراهیمی[3]

1- دانشجوی کارشناسی ارشد، مهندسی مکانیک، دانشگاه یزد، یزد
2- دانشیار، مهندسی مکانیک، دانشگاه یزد، یزد
3- استادیار، مهندسی مکانیک، دانشگاه یزد، یزد
* یزد، صندوق پستی 741-89195، rafeeyan@yazd.ac.ir

اطلاعات مقاله

مقاله پژوهشی کامل
دریافت: 10 مهر 1393
پذیرش: 20 آذر 1393
ارائه در سایت: 06 دی 1393

کلید واژگان:
ربات موازی سه درجه آزادی
تعقیب مسیر
الگوریتم بهینه‌سازی فاخته
الگوریتم بهینه‌سازی ژنتیک
کنترل مود لغزشی
منحنی اسپلاین

چکیده

در این تحقیق روشی جهت کنترل ربات موازی 3-RPR بر روی مسیری با وجود موانع متعدد در فضای کاری ارائه می‌گردد. برای این منظور، مسیری هموار و با کمترین طول که برخوردی با موانع موجود در فضای کاری ربات نداشته باشد با استفاده از روش منحنی‌های اسپلاین و الگوریتم بهینه‌سازی فاخته ایجاد شده و به عنوان مسیر مرجع تعقیب ربات مورد استفاده قرار می‌گیرد. برای بررسی سرعت و دقت همگرایی الگوریتم بهینه-سازی فوق، فرآیند بهینه‌سازی با الگوریتم ژنتیک نیز تکرار شده و روند همگرایی مقایسه می‌شود. در ادامه از روش کنترل مقاوم مود لغزشی برای کنترل این ربات در تعقیب مسیر مرجع استفاده می‌شود. انتخاب این کنترل‌کننده بر اساس نامعین بودن تعدادی از پارامترهای دینامیکی این ربات صورت می‌گیرد. نتایج حاصل از قسمت طراحی مسیر بهینه با توجه به سرعت و دقت بالای همگرایی، عملکرد مناسب الگوریتم بهینه‌سازی فاخته را به خوبی نشان می‌دهد. با توجه به نتایج حاصل مشاهده شد که مسیر طراحی شده با استفاده از روش منحنی اسپلاین بدون برخورد با موانع تعریف شده تا حد امکان کوتاه شده است. علاوه بر این، نتیجه می‌شود که روش کنترل مود لغزشی با وجود نامعینی‌های در نظر گرفته شده، قادر به کنترل ربات در تعقیب مسیر مرجع ایجاد شده با دقت بسیار خوبی است.

Sliding mode control of 3-RPR parallel robot on the optimal path using cuckoo optimization algorithm

Seyed Alireza Moezi, Mansour Rafeeyan*, Saeed Ebrahimi

Department of Mechanical Engineering, Yazd University, Yazd, Iran
*P.O.B. 89195-741Yazd, Iran,rafeeyan@yazd.ac.ir

ARTICLE INFORMATION

Original Research Paper
Received 02 October 2014
Accepted 11 December 2014
Available Online 27 December 2014

Keywords:
Three degrees of freedom robot
path generating
Cuckoo Optimization Algorithm
Genetic Optimization Algorithm
Sliding Mode Control
Spline Curve

ABSTRACT

The present study aims to implement an approach for trajectory control of a 3-RPR parallel manipulator over a path with obstacles in the workspace. For this purpose, using the spline curves approach and based on the cuckoo optimization algorithm, a smooth reference trajectory with minimum length is generated in the workspace to avoid robot collision with obstacles. The performance and accuracy of the cuckoo optimization algorithm in converging to the optimal solution is then compared with the Genetic algorithm. In the next step, the robust sliding mode control technique is adopted for trajectory control of the robot in the presence of some uncertainties. These uncertainties usually include the lengths and mass of the robot's links. The obtained results confirm the demanded level of performance and accuracy of the cuckoo optimization algorithm. It is also observed that the optimal trajectory with minimum length is generated using the spline curves approach. In addition, it is concluded that based on the sliding mode control technique, the robot can follow the desired trajectory very precisely in spite of the presence of the uncertainties in length and mass of the robot's links.

1- مقدمه

امروزه تولید انبوه در صنعت نیاز به سرعت و دقت بالا دارد. به همین منظور بحث رباتیک و مسائل پیرامون آن به یک موضوع جذاب برای محققان تبدیل شده است. در گذشته ربات‌های سری بخش‌های گسترده‌ای از صنایع را تشکیل می‌دادند. این ربات‌ها به دلیل سرعت نسبتاً پایین در انجام کارها و خطای ایجاد شده در تک‌تک اعضا به مرور جای خود را به ربات‌هایی با سرعت و دقت بالا (ربات‌های موازی) دادند.

یک ربات موازی را می‌توان این گونه تعریف کرد: مکانیزمی حلقه بسته که در آن مجری نهایی به وسیله‌ی حداقل دو حلقه‌ی سینماتیکی بسته به پایه متصل شده است [1]. با این وجود مرور مقالات نشان می‌دهد که هرگاه مکانیزمی به منظور جابجا کردن مواد، قطعات، ابزار یا وسایل خاص از طریق حرکات برنامه‌ریزی شده متغیر برای انجام گستره وسیعی از کارها و وظایف

برای ارجاع به این مقاله از عبارت ذیل استفاده نمایید:

Please cite this article using:
S. A. Moezi, M. Rafeeyan, S. Ebrahimi, Sliding mode control of 3-RPR parallel robot on the optimal path using cuckoo optimization algorithm, *Modares Mechanical Engineering*, Vol. 15, No. 2, pp. 147-158, 2015 (In Persian)

[Downloaded from mme.modares.ac.ir on 2024-10-12]

[DOR: 20.1001.1.10275940.1394.15.2.32.0]

طراحی شود، عبارت ربات برای آن استفاده می‌شود. در این نوع ربات‌ها به دلیل شباهت اکثر اعضای ربات به هم، هزینه تولید انبوه و تعویض قطعه معیوب کاهش می‌یابد و همچنین مشکلات ربات‌های سری از جمله خطای انباشته مفاصل و اعضا و سرعت پایین را ندارند.

در سال 1813 میلادی، اولین مکانیزم موازی که هشت‌وجهی[1] نامیده می‌شد توسط کاوچی مورد بررسی قرار گرفت. پس از آن در سال 1947 مک‌کاف مکانیزمی مشابه هشت‌وجهی ساخت و برای آزمایش تایرها در موقعیت‌های گوناگون از آن استفاده کرد. این مکانیزم در سال 1965 میلادی توسط استورات برای شبیه‌سازی پرواز توسعه داده شد. یک دسته از ربات‌های موازی، ربات‌های موازی صفحه‌ای هستند که امروزه در صنایع کاربرد فراوانی دارند و می‌توان از انواع آن‌ها به 3-RRR، 3-RPR، 3-PRR اشاره کرد که هر کدام از سه حلقه سینماتیکی بسته موازی تشکیل شده‌اند. امروزه محققان زیادی بر روی این دسته از ربات‌ها تحقیق و پژوهش انجام داده‌اند و از جمله کارهایی که در زمینه دینامیک و کنترل ربات‌های سه درجه آزادی موازی صورت پذیرفته است می‌توان از رن و همکارانش [2] که یک الگوریتم تطبیقی برای کنترل مسیر حرکت یک ربات موازی 3-PRR پیشنهاد دادند، نام برد. کردجزی و اکبرزاده [3] به کنترل ربات موازی 3-PRR با استفاده از روش گشتاور محاسبه شده پرداختند. این روش یک حالت خاص از روش خطی‌سازی با فیدبک سیستم‌های غیرخطی محسوب می‌شود. آن‌ها در این تحقیق کنترل‌کننده‌ی طراحی شده را به ازای ورودی‌های شیب و پله مورد آزمایش قرار دادند. نبوی و همکارانش [4] روشی برای حل معادلات دینامیکی ربات 3-PRR ارائه دادند. آن‌ها ابتدا معادلات سینماتیکی و دینامیکی ربات فوق را استخراج کردند. در این روش معادلات مقید بدست آمده از روش لاگرانژ طی فرایندی حذف شدند که این خود باعث ساده‌تر شدن معادلات دینامیکی این ربات می‌شد. در نهایت یک کنترل‌کننده‌ی PD[2] برای ربات مورد بحث طراحی کردند. ژانگ و همکارانش [5] به بررسی و کنترل ارتعاشات فعال ربات 3-PRR پرداختند. آن‌ها در تحقیق خود سه بازوی میانی ربات فوق را انعطاف‌پذیر در نظر گرفته و سپس به استخراج مودهای ارتعاشی بازوها به وسیله یک مبدل[3] پرداختند. آن‌ها همچنین[6] سه بازوی میانی ربات فوق را انعطاف‌پذیر در نظر گرفتند و سپس با استفاده از روش تقریبی مودهای فرضی، مودهای ارتعاشی را استخراج و با معادلات دینامیکی حاصل از روش لاگرانژ ترکیب کردند. در این تحقیق از روش FFT[4] برای آنالیز مودهای ارتعاشی استفاده شده است. سپس نتایج حاصل از تحقیق تئوری با نتایج حاصل از آزمون مودال نمونه عملی مقایسه گردید. اکبرزاده و همکارانش [7] کنترل‌کننده‌ای مقاوم به نامعینی و اغتشاشات سیستم طراحی کردند. در این روش نیازی به استخراج دقیق معادلات سیستم نبود. اسحاقیه و همکارانش [8] به بررسی معادلات دینامیکی ربات موازی 3-RPR پرداختندکه در این تحقیق سه بازوی میانی ربات موازی فوق را انعطاف‌پذیر فرض کرده و با استفاده از تئوری تیر اویلر- برنولی و روش لاگرانژ معادلات دینامیکی ربات موازی مزبور را استخراج کردند. آن‌ها همچنین به تأثیر انعطاف‌پذیری در مسیر حرکت مجری نهایی پرداختند.

اما به طور خاص در مورد کنترل ربات RPR با سه زنجیره سینماتیکی یعنی 3-RPRکه موضوع این تحقیق است تحقیقات زیادی صورت نگرفته است. تنها مورد قابل اشاره کار ویلیام در سال 1999 [9] است. ویلیام به طراحی، ساخت و کنترل ربات موازی با سه حلقه سینماتیکی RPR پرداخت. او در تحقیق خود به بررسی سینماتیک معکوس، مستقیم و فضای کاری ربات موازی فوق پرداخت. او از سه بازوی پنوماتیکی پیشران در طرح خود استفاده کرد و برای هر بازو یک حلقه‌ی بسته کنترلی فیدبک‌دار در نظر گرفت. در این طرح از حسگر جابجایی[5] برای اندازه‌گیری مقدار جابه‌جایی بازوها و از سه شیر برقی[6] برای تنظیم فشار اعمالی روی بازوها استفاده شده است. برای کنترل میزان جابجایی بازوهای ربات از یک کنترل‌کننده‌ی PID[7] که ورودی آن تفاضل خروجی حسگر جابه‌جایی و مطلوب مورد نظر است و خروجی آن ولتاژ اعمالی به شیرهای برقی است، استفاده شده است.

یکی از مسائل مهمی که هر ربات موازی باید در گذر مسیر خود از نقطه شروع به نقطه هدف از آن بهره‌مند باشد جلوگیری از برخورد با موانع و همچنین ایجاد مسیری کوتاه تا هدف است. در این تحقیق این موضوع به عنوان هدف تولید مسیر در نظر گرفته شده است، از جمله روش‌هایی که برای تولید مسیر استفاده می‌شود:

1- تبدیل منحنی به خط های به هم پیوسته
2- تبدیل منحنی به یک معادله ریاضی پارامتریک و تعیین ضرایب آن توسط روش‌های بهینه سازی (چند جمله‌ای‌ها)
3- استفاده از شبکه‌های عصبی و منطق فازی در تولید مسیر

بر همین اساس از جمله تحقیقاتی که در زمینه طراحی مسیر ربات‌های موازی صفحه‌ای صورت پذیرفته است، پارسا و همکارانش [10] به طرح ریزی مسیر برای یک ربات سه درجه آزادی 3-RRR با استفاده از شبکه عصبی با حضور مانع در مسیر پرداختند. دانش و اباذری [11] مسیری برای ربات موازی 3-RRR با استفاده از الگوریتم ژنتیک طراحی کردند. مسیر تولیدی توانایی عبور از موانع را داشت. ناسا و باندیویدهیای [12] مسیری برای عبور از نقاط تکین ربات موازی 3-RRR طراحی کردند. لاهور و همکارانش [13] برای ربات موازی 3-RRR مسیری بهینه طراحی کردند. آن‌ها در این تحقیق نقاط تکین ربات موازی را به عنوان مانع در نظر گرفتند.

در این تحقیق از منحنی اسپلاین[8] [14] که می‌توان گفت ترکیبی از موارد اشاره شده 1 و 2 است برای تولید مسیر استفاده شده است. یکی از مزیت‌های خوب این منحنی در این تحقیق، امکان ایجاد مسیرهایی با اشکال متفاوت در صفحه است. به طوری که می‌توان با تغییر تعداد و مقدار نقاط کنترل منحنی اسپیلاین، مسیرهای متنوعی ایجاد کرد و بیشتر مسیرهای موجود بین نقاط ابتدا و انتها را پوشش داد. در این روش مکان نقاط کنترل منحنی که تعداد آن متناسب با طول و هموار بودن منحنی برگزیده می‌شود توسط الگوریتم بهینه‌سازی فاخته[9] که توسط رجبیون [15] ارائه شد، طوری تعیین می‌گردد که مسیر دارای کوتاهترین فاصله باشد و ربات موازی فوق به موانع موجود در مسیر برخورد نکند. در این تحقیق تابع هدف به صورت ترکیبی از طول مسیر و قیدهای برخورد با موانع ایجاد شده است. همچنین برای انتخاب نقاط از مش‌بندی فضای دو بعدی که سبب ایجاد خطا می‌گردد استفاده نشده و نقاط به صورت پیوسته در محیط انتخاب می شوند. در ادامه برای مقایسه سرعت و دقت همگرایی الگوریتم فوق، نتایج همگرایی این الگوریتم با الگوریتم مشهور ژنتیک مقایسه می‌گردد. همچنین نتایج همگرایی این الگوریتم توسط روشی صحه‌گذاری خواهد شد. سپس مسیر فوق با

1- Octahedral
2- Proportional-Derivative Controller
3- Transducer
4- Fast Fourier Transform

5- LVDT Sensor
6- Solenoid Valve
7- Proportional-Integral-Derivative Controller
8- Spline
9- Cuckoo Optimization Algorithm (COA)

[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]

استفاده از روشی که در ادامه تحقیق بیان شده است به عنوان موقعیت مرجع به ربات موازی مورد نظر داده خواهد شد. در ادامه برای اینکه ربات موازی صفحه‌ای بتواند موقعیت مرجع را رهگیری نماید کنترل‌کننده‌ای از نوع مود لغزشی که به نامعینی‌های پارامتری مقاوم است طراحی می‌گردد. در شکل 1 شماتیکی از مراحل انجام آورده شده است.

2- سینماتیک ربات سه درجه آزادی

با توجه به هندسه‌ی شکل 2 برای هر زنجیره سینماتیکی می‌توان رابطه برداری (1) را نوشت:

$$A_iB_i + B_iC_i = A_iP + PC_i \quad (1)$$

مختصات نقطه C_i با استفاده از زنجیره‌ی $A_iB_iC_i$ می‌تواند به صورت روابط (2) و (3) نوشته شود:

$$x_{ci} = x_{ai} + (\rho_i + l)\cos(\theta_i) \quad (2)$$

$$y_{ci} = y_{ai} + (\rho_i + l)\sin(\theta_i) \quad (3)$$

که x_{ci} و y_{ci} مختصات نقطه C_i، و x_{ai} و y_{ai} مختصات ثابت نقطه A_iهستند. l طول بازوهای میانی صلب، B_iC_i، است.

مختصات نقطه C_i با استفاده از مختصات سکوی متحرک نیز می‌تواند به صورت روابط (4) و(5) نوشته شود:

$$x_{ci} = x_p + x_{ci}^0\cos(\phi_p) - y_{ci}^0\sin(\phi_p) \quad (4)$$

$$y_{ci} = y_p + x_{ci}^0\sin(\phi_p) - y_{ci}^0\cos(\phi_p) \quad (5)$$

که x_p و y_p مختصات مرکز جرم سکوی متحرک، نقطه P، نسبت به دستگاه ثابت وϕ_p زاویه‌ی سکوی متحرک نسبت به محور x است. x_{ci}^0و y_{ci}^0 مختصات ثابت نقطه C_i هستند، که از مرکز جرم سکوی متحرکP وقتی $\phi_p = 0$ است اندازه‌گیری می‌شوند.

با جایگذاری معادلات (2) در (4) و (3) در (5) و مرتب کردن آن‌ها معادلات قید بدست می‌آیند:

$$\Gamma_{2i-1} \equiv x_{ai} + (\rho_i + l)\cos(\theta_i) - x_p - x_{ci}^0\cos(\phi_p) + y_{ci}^0\sin(\phi_p) = 0 \quad (6)$$

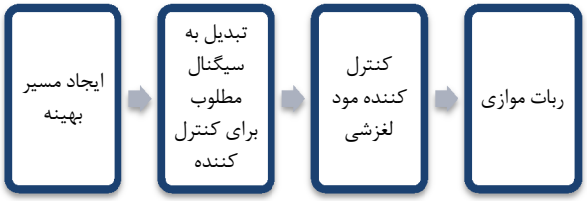
$$\Gamma_{2i} \equiv y_{ai} + (\rho_i + l)\sin(\theta_i) - y_p - x_{ci}^0\sin(\phi_p) + y_{ci}^0\cos(\phi_p) = 0 \quad (7)$$

Γ_{2i-1}معادلات قید در راستای x و Γ_{2i} معادلات قید در راستایy هستند. با توجه به اینکه x_{ci}^0 و y_{ci}^0 مقادیری ثابت‌اند و به حرکت مجری نهایی وابسته نیستند می‌توان این دو را، با توجه به شکل 3 که مجری نهایی را نشان می‌دهد، به صورت روابط (8) و (9) مرتب کرد:

$$x_{ci}^0 = r\cos(\phi_i) \quad (8)$$

$$y_{ci}^0 = r\sin(\phi_i) \quad (9)$$

همانطورکه در شکل 3 مشخص است؛ r برابر طول پاره‌خط PC_i است که مرکز سطح مجری نهایی، نقطه‌ی P، را به مفاصل C_i وصل می‌کند و ϕ_i زاویه‌ی r با محور x است. با توجه به اینکه مجری نهایی یک مثلث متساوی‌اضلاع است می‌توان r را برحسب طول ضلع مثلث، نقطه‌ی b، به دست آورد:



شکل 1 شماتیک روند ایجاد مسیر بهینه و تبدیل آن به سیگنال مطلوب

$$r = PC_i = \frac{2}{3}b\cos(30) = \frac{\sqrt{3}}{3}b \quad (10)$$

همچنین از هندسه مثلث متساوی الاضلاع، زوایای ϕ_i به شرح رابطه (11) است:

$$\phi_1 = -\frac{\pi}{6}; \phi_2 = \frac{\pi}{2}; \phi_3 = \frac{7\pi}{6} \quad (11)$$

با جایگذاری معادلات (8) و (9) در (6) و (7) و ساده‌سازی آن‌ها به معادلات (12) و (13) می‌رسیم:

$$\Gamma_{(2i-1)} \equiv x_{ai} + (\rho_i + l)\cos(\theta_i) - x_p - r\cos(\phi_i - \phi_p) = 0 \quad (12)$$

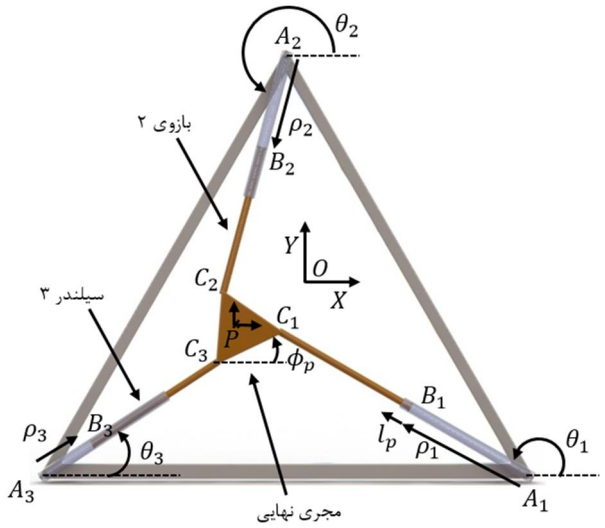
$$\Gamma_{(2i)} \equiv y_{ai} + (\rho_i + l)\sin(\theta_i) - y_p - r\sin(\phi_i - \phi_p) = 0 \quad (13)$$

رابطه (12) شامل سه معادله‌ی قید در راستای x و رابطه (13) شامل سه معادله‌ی قید در راستای yاست. این شش معادله یک دستگاه معادلات جبری را تشکیل می‌دهند که بایستی با هم حل شوند. در صورت حل دستگاه فوق بر حسب پارامترهای x_p, y_p, ϕ_p به معادلات (14) و (15) خواهیم رسید:

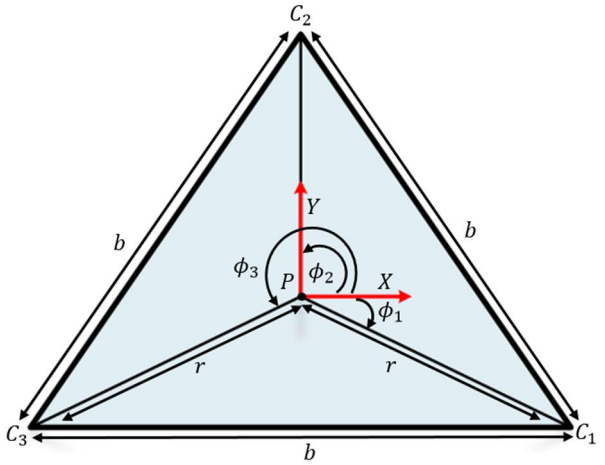
$$\theta_i = \tan^{-1}\left(\frac{y_p(t) + r\sin(\phi_i + \phi_p(t)) - ya_i}{x_p(t) + r\cos(\phi_i + \phi_p(t)) - xa_i}\right) \quad (14)$$

$$\rho_i = \frac{x_p(t) + r\cos(\phi_i + \phi_p(t)) - xa_i}{\cos(\theta_i(t))} - l \quad (15)$$

در تمام معادلات سینماتیکی ربات $i = 1,2,3$ است.



شکل 2 ساختار ربات3-RPR



شکل 3 مجری نهایی ربات

[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]

3- استخراج معادلات حرکت

ربات موازی سه درجه آزادی فوق از سه حلقه سینماتیکی بسته RPR تشکیل شده است. در این ربات تمام بازوهای حلقه‌های سینماتیکی ربات صلب بوده و چون ربات در صفحه حرکت می‌کند از انرژی پتانسیل آن صرفنظر شده است. همچنین از جرم پیستون‌های متصل به بازوها به علت اینکه در برابر جرم بازوها ناچیز است صرفنظر می‌شود.

برای استخراج معادلات دینامیکی ربات موازی فوق از روش لاگرانژ استفاده می‌شود. بنابراین معادلات انرژی جنبشی ربات موازی صفحه‌ای فوق برای جایگذاری در معادلات لاگرانژ به شرح روابط (16) تا (19) استخراج می‌شوند:

$$T_C = \sum_{i=1}^{3} \frac{1}{2}(I_C + I_p)\dot{\theta}_i^2 \quad (16)$$

$$T_L = \sum_{i=1}^{3} \frac{1}{2} m_a \left[\dot{\rho}_i^2 + \left((l_p + \rho_i)\dot{\theta}_i\right)^2\right] \quad (17)$$

$$T_P = \frac{1}{2} m_p \left(\dot{x}_p + \dot{y}_p\right)^2 + \frac{1}{2} I_e \dot{\phi}_p^{\,2} \quad (18)$$

$$T = T_C + T_L + T_P \quad (19)$$

T_C, T_L, T_P به ترتیب انرژی‌های جنبشی سیلندرها، بازوهای میانی و سکوی متحرک هستند. در روابط بالا I_p ممان اینرسی جرمی هر کدام از سه بازوی میانی حول محور عمود بر صفحه‌ی ربات موازی، I_C ممان اینرسی جرمی هر کدام از سه سیلندر حول محور عمود بر صفحه‌ی ربات موازی در نقطه A_i و $\dot{\theta}_i$ سرعت زاویه‌ای سیلندر i ام، m_a جرم هر بازوی میانی، l_p فاصله بین مرکز جرم پیستون و بازوی میانی، ρ_i فاصله‌ی مرکز جرم پیستون B_i از مفصل A_i، $\dot{\rho}_i$ سرعت خطی پیستون در راستای محور بازوی میانی i ام، m_p جرم سکوی متحرک، I_e ممان اینرسی جرمی سکو حول محور عمود بر صفحه‌ی سکو در مرکز جرم P، $\dot{x}_p, \dot{y}_p$ به ترتیب سرعت خطی مرکز جرم P در راستای محورهای x و y، و $\dot{\phi}_p$ سرعت زاویه‌ای مجری نهایی حول نقطه P است.

در این تحقیق ربات موازی صفحه‌ای فوق با استفاده از سه مختصه‌ی x_p, y_p, ϕ_p مدل‌سازی شده است، بدین منظور در روابط انرژی جنبشی به جای مختصه‌های $\rho = [\rho_1 \rho_2 \rho_3]$ ، $\theta = [\theta_1 \theta_2 \theta_3]$ و $\dot{\rho}_i, \dot{\theta}_i, i = 1,2,3$ از مختصه‌های مذکور با استفاده از روابط سینماتیک (14) و (15) و مشتقات آن‌ها استفاده شده است.

بنا بر روابط بالا تابع لاگرانژین به صورت رابطه (20) است:

$$L = T \quad (20)$$

با استفاده از فرمول اویلر لاگرانژ معادلات دینامیکی سیستم به صورت روابط (21) تا (23) استخراج می‌گردد:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_p}\right) - \frac{\partial L}{\partial x_p} = b_1 F_1 + b_2 F_2 + b_3 F_3 + F_4 \quad (21)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}_p}\right) - \frac{\partial L}{\partial y_p} = b_4 F_1 + b_5 F_2 + b_6 F_3 + F_5 \quad (22)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}_p}\right) - \frac{\partial L}{\partial \phi_p} = b_7 F_1 + b_8 F_2 + b_9 F_3 + M_1 \quad (23)$$

که b_i , $i = 1,2,\dots,9$ ضرایب نیروهای وارد شده به مختصه‌های نهایی از طرف عملگرهای کشویی به شرح روابط (24) تا (26) است:

$$b_i = \frac{\sigma_1}{\sigma_2} \qquad i = 1,2,3 \quad (24)$$

$$b_i = \frac{\sigma_3}{\sigma_4} \qquad i = 4,5,6 \quad (25)$$

$$b_i = \frac{\sigma_5}{\sigma_6} \qquad i = 7,8,9 \quad (26)$$

که σ_1 تا σ_6 صورت روابط (27) تا (32) نوشته می‌شوند:

$$\sigma_1 = x_p - xa_i + r\cos(\phi_i + \phi_p) \quad (27)$$

$$\sigma_2 = \left((xa_i - x_p)^2 + (ya_i - y_p)^2 - 2r\left(\cos(\phi_i + \phi_p)(xa_i - x_p) + \sin(\phi_i + \phi_p)(ya_i - y_p)\right) + r^2\right)^{\frac{1}{2}} \quad (28)$$

$$\sigma_3 = y_p - ya_i + r\sin(\phi_i + \phi_p) \quad (29)$$

$$\sigma_4 = \left((xa_i - x_p)^2 + (ya_i - y_p)^2 - 2r\left(\cos(\phi_i + \phi_p)(xa_i - x_p) + \sin(\phi_i + \phi_p)(ya_i - y_p)\right) + r^2\right)^{\frac{1}{2}} \quad (30)$$

$$\sigma_5 = -r\left(\cos(\phi_i + \phi_p)(ya_i - y_p) - \sin(\phi_i + \phi_p)(xa_i - x_p)\right) \quad (31)$$

$$\sigma_6 = \left((xa_i - x_p)^2 + (ya_i - y_p)^2 - 2r\left(\cos(\phi_i + \phi_p)(xa_i - x_p) + \sin(\phi_i + \phi_p)(ya_i - y_p)\right) + r^2\right)^{\frac{1}{2}} \quad (32)$$

که در روابط (21) تا (23) F_3, F_2, F_1 به ترتیب نیروهای وارد شده به سه عملگر کشویی ربات و M_1, F_5, F_4 نیروها و گشتاور خارجی وارد شده به مجری نهایی در جهت مختصه‌های تعمیم یافته یعنی x_p, y_p, ϕ_p هستند.

متغیرهای حالت سیستم بصورت رابطه (33) تعریف می‌شوند:

$$q = \begin{bmatrix} x_p \\ y_p \\ \phi_p \end{bmatrix} \quad (33)$$

معادلات (21) تا (23) را می‌توان به فرم (34) نوشت:

$$M(q)\ddot{q} + C(q,\dot{q}) = H(q)U + F_{ex} \quad (34)$$

که M ماتریس اینرسی، C بردار کوریولیس، $H(q)$ ماتریس آرایش ورودی‌ها و بردارهای F_{ex} و U برابرند با:

$$U = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}; F_{ex} = \begin{bmatrix} F_4 \\ F_5 \\ M_1 \end{bmatrix} \quad (35)$$

اگر رابطه (34) را به فرم فضای حالت بنویسیم خواهیم داشت:

$$\ddot{q} = M^{-1}\left(H(q)U + F_{ex} - C(q,\dot{q})\right) \quad (36)$$

$$x_1 = x_p, x_2 = \dot{x}_p, x_3 = y_p, x_4 = \dot{y}_p, x_5 = \phi_p, x_6 = \dot{\phi}_p \quad (37)$$

$$\dot{x}_1 = x_2, \dot{x}_2 = \ddot{x}_p, \dot{x}_3 = x_4, \dot{x}_4 = \ddot{y}_p, \dot{x}_5 = x_5, \dot{x}_6 = \ddot{\phi}_p \quad (38)$$

4- طراحی کنترل‌کننده مود لغزشی

عدم دقت در مدل ممکن است ناشی از عدم دقت در تعیین پارامترهای یک سیستم دینامیکی یا نمایش ساده شده‌ی هدف دار یک مدل دینامیکی باشد. در علم کنترل به دسته‌ی اول نامعینی‌های پارامتری و به دسته‌ی دوم نامعینی‌های دینامیکی گویند. یک روش کارمد برای کنترل مقاوم این گونه سیستم‌ها روش کنترل مود لغزشی می‌باشد. عدم دقت در مدل‌سازی می‌تواند اثرات نامطلوب شدیدی بر سیستم‌های غیر خطی بگذارد. بنابراین در هر طرحی باید اثرات ناشی از عدم قطعیت‌ها در نظر گرفته شود.

با توجه به مراجع [16-17] برای یک سیستم دینامیکی تک ورودی می‌توان نوشت:

$$x^{(n)} = f(x) + b(x)u \quad (39)$$

که در آن اسکالر u ورودی کنترل، $x^{(n)}$ مشتق n ام x ، $f(x), b(x)$ عبارات همراه با نامعینی و $x = [x\ \dot{x} \dots x^{n-1}]^T$ بردار حالت است. مساله کنترلی فوق بدین صورت است که حالت x را چنان بیابیم که یک حالت متغیر با زمان

[Downloaded from mme.modares.ac.ir on 2024-10-12]

[DOR: 20.1001.1.10275940.1394.15.2.32.0]

مشخص $x_d = [x_d\, \dot{x}_d \dots x_d^{n-1}]^T$ را با وجود نامعینی در $f(x)$ و $b(x)$ تعقیب کند. خطای تعقیب مسیر در متغیر حالت x را به صورت $\tilde{x} = x - x_d$ تعریف کرده و فرض می‌کنیم:

$$\tilde{x} = x - x_d = \left[\tilde{x}, \dot{\tilde{x}}, \dots, \tilde{x}^{(n-1)}\right] = \left[x - x_d, \dot{x} - \dot{x}_d, \dots, x^{(n-1)} - x_d^{(n-1)}\right]^T \quad (40)$$

حال اگر یک سطح متغیر با زمان $s(t)$ در فضای حالت R^n با معادله اسکالر $s(x,t)$ تعریف کنیم:

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x} = 0 \quad (41)$$

که λ یک ثابت اکیدا مثبت است. برای کم کردن خطای حاصل از تعقیب مسیر می توان عبارت انتگرالی را به معادله (41) به صورت رابطه (42) اضافه کرد:

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \left(\int_0^t \tilde{x} dt\right) = 0 \quad (42)$$

هدف اصلی از این گونه ارائه، نگه داشتن شیب سطح لغزش $s(x,t)$ نزدیک به صفر است. بنابراین یکی از روش‌های معمول، پیدا کردن ورودی U خارج از سطح لغزش $s(x,t)$ است.

$$\frac{1}{2}\left(\frac{d}{dt}\right) s^2(x,t) \le -\zeta |s(x,t)| \quad (43)$$

که ζ یک ثابت مثبت می‌باشد.

اگر: $S(0) > 0 \rightarrow \frac{d}{dt} S(t) \le -\zeta$ (44)

برای حذف ترم مشتقی، از یک ترم انتگرالی از $t = 0$ تا $t = t_{\text{reach}}$ استفاده شده است.

$$\int_{t=0}^{t=t_{\text{reach}}} \frac{d}{dt} s(t) \le -\int_{t=0}^{t=t_{\text{reach}}} \eta \rightarrow s(t_{\text{reach}}) - s(0) \le -\zeta(t_{\text{reach}} - 0) \quad (45)$$

جایی که t_{reach} برابر است با زمانی که منحنی به سطح لغزش می‌رسد فرض می‌شود $s(t_{\text{reach}} = 0)$ بصورت روابط (46) و (47) تعریف شده است:

$$0 - s(0) \le -\eta(t_{\text{reach}}) \rightarrow t_{\text{reach}} \le \frac{s(0)}{\zeta} \quad (46)$$

و

اگر: $s(0) < 0 \rightarrow 0 - s(0) \le -\eta(t_{\text{reach}}) \rightarrow s(0) \le -\zeta(t_{\text{reach}}) \rightarrow t_{\text{reach}} \le \frac{|s(0)|}{\eta}$ (47)

معادله (47) تضمین می‌کند مدت زمانی که به سطح لغزش می‌رسد کمتر از $\frac{|s(0)|}{\zeta}$ از جایی که منحنی خارج از $s(t)$ است:

اگر: $s_{t_{\text{reach}}} = s(0) \rightarrow \text{error}(x - x_d) = 0$ (48)

فرض می‌شود s به فرم (49) تعریف شود:

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)\tilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \quad (49)$$

مشتق s که $\dot{s}$ می‌باشد به صورت رابطه (50) تعریف می‌گردد:

$$\dot{s} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \quad (50)$$

حال اگر فرض شود که سیستم مرتبه 2 به صورت رابطه (51) باشد:

$$\ddot{x} = f + u \rightarrow \dot{s} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \quad (51)$$

که f نامعینی دینامیکی است. همچنین $s, \dot{s}$ برابر صفر می‌باشند. بنابراین $\hat{U}$ بصورت رابطه (52) تعریف می‌شود:

$$\hat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \quad (52)$$

یک راحل ساده برای تعیین شرایط سطح لغزش، زمانی که پارامترهای سیستم نامعین هستند، قانون کنترلی سوئیچینگ است:

$$U_{dis} = \hat{U} - K(x,t) \cdot \text{sgn}(s) \quad (53)$$

که تابع سوئیچینگ $\text{sgn}(s)$ برابر است با:

$$\text{sgn}(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \quad (54)$$

و $K(x,t)$ یک ثابت مثبت است. با توجه به معادله (46) می‌توان به معادله‌ای به فرم (55) رسید:

$$\frac{1}{2}\frac{d}{dt} s^2(x,t) = \dot{s} \cdot s = \left[f - \hat{f} - K\text{sgn}(s)\right] \cdot S = (f - \hat{f}) \cdot S - K|S| \quad (55)$$

با جایگذاری معادله‌ی (50) در (49) داریم:

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^2 \left(\int_0^t \tilde{x} dt\right) = (\dot{x} - \dot{x}_d) + 2\lambda(\dot{x} - \dot{x}_d) - \lambda^2(x - x_d) \quad (56)$$

بنابراین U به صورت رابطه (57) تقریب زده می‌شود:

$$\hat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{x} - \dot{x}_d) + \lambda^2(x - x_d) \quad (57)$$

بدین ترتیب قانون کنترل مود لغزشی را برای یک ربات چند درجه آزادی موازی می‌توان بصورت روابط (58) تا (61) تعمیم داد و استخراج کرد:

$$H(q)\tau = \tau_{eq} + \tau_{dis} \quad (58)$$

که τ_{eq} به صورت رابطه (59) محاسبه می‌گردد:

$$\tau_{eq} = \left[C(q,\dot{q}) + M \cdot \dot{S} - F_{ex}\right] \quad (59)$$

و τ_{dis} برابر است با :

$$\tau_{dis} = M \cdot K \cdot \text{sgn}(s) \quad (60)$$

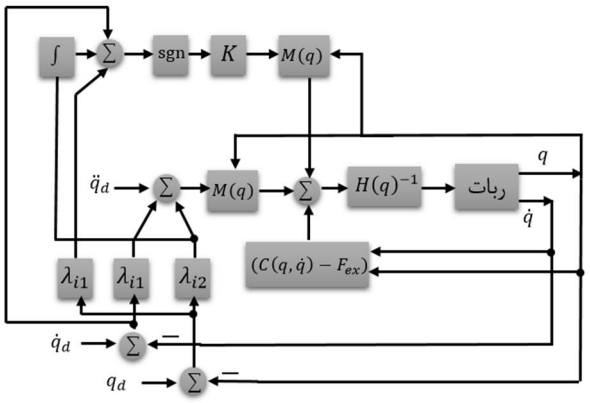
با جایگذاری فرمول‌های (59) و (60) در (58) خروجی کنترل‌کننده به صورت (61) محاسبه می‌گردد:

$$\tau = H(q)^{-1}\left[M \cdot \dot{S} + C(q,\dot{q}) - F_{ex} + M \cdot K \cdot \text{sgn}(s)\right] \quad (61)$$

همچنین روند طراحی کنترل‌کننده در شکل 4 رسم شده است.

5- بهینه سازی و ایجاد مسیر کوتاه بدون برخورد

برای حرکت بدون برخورد از نقطه شروع f_{in} به نقطه هدف f_{end} در مسیر دو بعدی، انتخاب کوتاه ترین مسیر، همواره یکی از چالش‌های تولید مسیر بوده است. در این تحقیق تعدادی نقطه در فضای دو بعدی برگزیده شده و به کمک روش ایجاد منحنی اسپلاین، مسیری از این نقاط و نقاط شروع و پایان می‌گذرد. مکان نقاط به گونه‌ای انتخاب می‌شوند که مسیری کوتاه و بدون برخورد را حاصل شود. بدین صورت که هر نقطه از محیطی پیرامون ربات موازی که متعلق به مانع باشد بسته به مقدار فاصله تا مرز مانع و محیط پیرامون مانع مقدار عددی به خود می‌گیرد که در بهینه سازی از این مقدار جهت جلوگیری از برخورد با موانع استفاده می‌شود. بنابراین تابع هدف به



شکل 4 روند طراحی کنترل‌کننده مود لغزشی

[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]

گونه‌ای تعیین شده است که مسیر حاصله کمترین طول را داشته باشد و همچنین به موانع بر خورد نکند. برای حل مسأله بهینه‌سازی طرح شده در این تحقیق از الگوریتم بهینه‌سازی فاخته استفاده شده است. این روش نسبت به الگوریتم‌هایی از قبیل ژنتیک و تجمع ذرات از سرعت و دقت همگرایی بالاتری برخوردار است [15].

در صورتی که تعداد n نقطه کنترل جهت تولید مسیر به روش ایجاد منحنی اسپلاین در نظر گرفته شده باشد و $f(n)$ برابر مقادیر عددی آن نقاط در بازه تعریف شده باشد. در این صورت می‌توان روابط مسیر تولید شده را به صورت (62) تا (65) بیان کرد:

$$t = [t_1 \;\; t_{\text{nop}}] \tag{62}$$

$$i = \left[0, \frac{1}{n+1}, \frac{2}{n+1}, \dots, \frac{n}{n+1} \;\; 1\right] \tag{63}$$

$$f_x(t) = \text{SPL}\left(i, [f_{x_{\text{in}}} \; f_{x_1} \; f_{x_2} \dots \; f_{x_n} \; f_{x_{\text{end}}}], \; t\right) \tag{64}$$

$$f_y(t) = \text{SPL}\left(i, [f_{y_{\text{in}}} \; f_{y_1} \; f_{y_2} \dots \; f_{y_n} \; f_{y_{\text{end}}}], \; t\right) \tag{65}$$

که $f_{x_{\text{in}}}, f_{x_{\text{end}}}$ به ترتیب نقاط ابتدا و انتهای مسیر در صفحه‌ی x و $f_{y_{\text{in}}}, f_{y_{\text{end}}}$ به ترتیب نقاط ابتدا و انتهای مسیر در صفحه‌ی y می‌باشند. همچنین متغیر nop تعداد و $(t_1 \dots t_{\text{nop}})$ مقدار عددی نقاط در نظر گرفته شده در بازه‌ی [0 1] برای رسم منحی است. که هر چه تعداد این نقاط بیشتر باشد منحنی رسم شده از دقت بیشتری برخوردار است.

(f_x, f_y) نقاط مسیر هستند که با تغییر t از t_1 تا t_{nop}، از نقطه شروع تا هدف تغییر می‌کنند. SPL تابعی است که از نقاط ورودی خود منحنی اسپلاین عبور می‌دهد و مقادیر را در هر نقطه بر می‌گرداند. این تابع در نرم-افزار متلب تعریف شده است. عملکرد این تابع در شکل 5 نشان داده شده است.

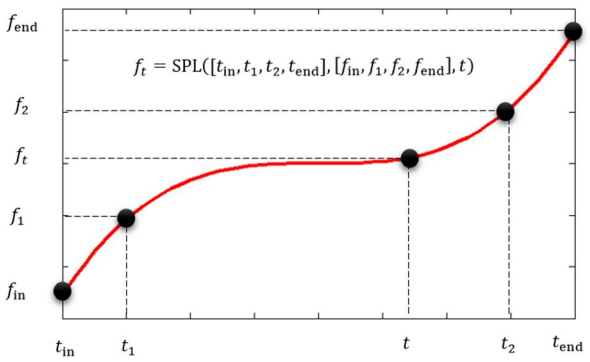
همواره برای بهینه سازی هر مسئله‌ای باید تابع هدف و متغیرهای بهینه سازی مشخص شوند. در این مسئله متغیرهای بهینه‌سازی نقاط f_{x_n}, f_{y_n} هستند. که همان نقاط کنترل منحنی اسپیلاین هستند. در این تحقیق شش نقطه کنترل، یعنی سه نقطه کنترل در جهت محور x و سه نقطه کنترل در جهت محور y در نظر گرفته شده‌اند. تابع هدف در این مسئله طوری مشخص شده است که کاهش آن به معنای کاهش طول مسیر و جلوگیری از برخورد با موانع است. بنابراین تابعی که نمایانگر موانع در فضای دوبعدی است با نام $OB(X,Y)$ تعریف شده که تابعی از مکان است.

طول مسیر به صورت روابط (66) تا (68) است:

$$Ln_x(t_i) = f_x(t_i) - f_x(t_{i-1}) \qquad i = 2{:}\,\text{nop} \tag{66}$$

$$Ln_y(t_i) = f_y(t_i) - f_y(t_{i-1}) \qquad i = 2{:}\,\text{nop} \tag{67}$$

$$L = \sum_{t=2}^{\text{nop}} \left(\left(Ln_x^2(t_i) + Ln_y^2(t_i) \right)^{\frac{1}{2}} \right) \qquad i = 2{:}\,\text{nop} \tag{68}$$



شکل 5 نحوه عملکرد تابع SPL

در این تحقیق تابع هدف به صورت رابطه (69) در نظر گرفته شده است:

$$\text{Cost Value} = L(1 + \beta \cdot OB) + \alpha \cdot OB \tag{69}$$

که L, β, α, OB به ترتیب طول مسیر، یک ثابت مثبت، یک ثابت مثبت، و تابع برخورد است که در این تحقیق ضرایب $\beta = 100$ ، $\alpha = 1000$ که ضرایب قیود برخورد مسأله هستند با چند بار حل مسأله بهینه‌سازی به نحوی انتخاب شدند که تابع هزینه که متشکل از طول مسیر و قیدهای برخورد با موانع دایره‌ای و مستطیلی است به شکل مطلوبی عمل کند و به سمت یکی از جمله‌های تشکیل دهنده تابع هزینه نظیر قیدها یا طول مسیر منحرف نگردد. در این مسأله قید برخورد از سه قید به شرح رابطه (70) تشکیل شده است:

$$OB = OB_1 + OB_2 + OB_3 \tag{70}$$

که OB_1, OB_2 به ترتیب توابع قید موانع دایره‌ای شکل و OB_3 تابع قید مانع مستطیلی است که به صورت روابط (71) تا (78) تعریف می‌شوند:

$$d_{c1} = \sqrt{(f_x(t_i) - x_{c1})^2 + (f_y(t_i) - y_{c1})^2} \tag{71}$$

$$D_{c1} = \max\left(1 - \frac{d_{c1}}{r_{c1}}, 0\right) \tag{72}$$

$$OB_{c1} = \text{mean}(D_{c1}) \tag{73}$$

$$d_{c2} = \sqrt{(f_x(t_i) - x_{c2})^2 + (f_y(t_i) - y_{c2})^2} \tag{74}$$

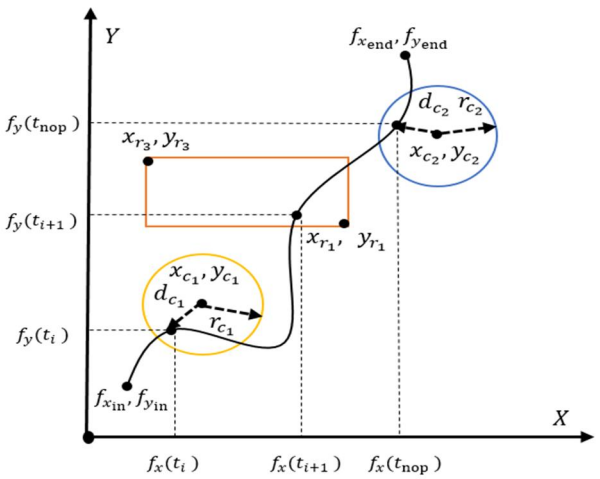
$$D_{c2} = \max\left(1 - \frac{d_{c2}}{r_{c2}}, 0\right) \tag{75}$$

$$OB_{c2} = \text{mean}(D_{c2}) \tag{76}$$

$$\begin{aligned} &J_1 = x_{r3} - f_x(t_i) \;\; ; \;\; J_2 = f_x(t_i) - x_{r1} \\ &J_3 = y_{r3} - f_y(t_i) \;\; ; \;\; J_4 = f_y(t_i) - y_{r1} \\ &J = (J_1 \cdot J_2) > 0 \;\&\; (J_3 \cdot J_4) > 0 \end{aligned} \tag{77}$$

$$OB_3 = \sum_{t=2}^{\text{nop}} J \tag{78}$$

که $i = 2{:}\,1{:}\,\text{nop}$، r_{c1}, r_{c2} به ترتیب شعاع دو دایره، $y_{r_3}, y_{r_1}, x_{r_3}, x_{r_1}$ به ترتیب مختصات مکانی دو رأس مستطیل در راستای x و y هستند. همچنین $y_{c_2}, y_{c_1}, x_{c_2}, x_{c_1}$ به ترتیب مختصات مکانی مراکز دو دایره در راستای x و y هستند. در این تحقیق اگر شرایط رابطه (77) ارضاء گردد J مقدار عددی یک و در غیر این صورت صفر را به خود می‌گیرد و با توجه به این موضوع اگر نقاط مسیر تولید شده به مانع مستطیلی برخورد کند مقدار عددی رابطه (78) افزایش و در غیر این‌صورت کاهش یافته تا به صفر برسد. در شکل 6 شماتیکی از روند مطرح شده به ازای چند نقطه از تمام نقاط مسیر آورده شده است. لازم به ذکر است که برای تمام نقاط تشکیل دهنده منحنی روابط (71) تا (78) چک می‌گردد.



شکل 6 روند مسیر تولید شده در حضور قیدهای برخورد

[Downloaded from mme.modares.ac.ir on 2024-10-12]

[DOR: 20.1001.1.10275940.1394.15.2.32.0]

همچنین در این تحقیق برای عدم برخورد مجموعه مکانیکی ربات موازی، نظیر بازوها به موانع، فرض شده است که ارتفاع مجموعه از صفحه زیاد باشد و موانع در ارتفاع پایین‌تری قرار داشته باشند. در ضمن برای عدم برخورد ابزار متصل شده به مجری نهایی، به موانع در این تحقیق، شعاع ابزار به ابعاد موانع واقعی درحل مسأله ، اضافه می‌گردد. بدین صورت محیط ابزار متصل شده به مجری نهایی ربات، به موانع برخورد نخواهد کرد.

6- الگوریتم بهینه‌سازی فاخته

الگوریتم بهینه‌سازی فاخته همانند سایر روش‌های بهینه‌سازی نظیر الگوریتم-های تجمع ذرات، ژنتیک، کلونی مورچه و سایر روش‌ها با یک جمعیت اولیه به نام فاخته شروع می‌گردد، هر فاخته تعدادی تخم دارد که آن‌ها را در لانه پرندگانی که تخم‌های آن‌ها از نظر شکل، رنگ و اندازه مشابه می‌باشد مخفی می‌کند، تعدادی از این تخم‌ها شروع به رشد کرده و به یک فاخته مادر تبدیل می‌گردد و بقیه توسط پرنده میزبان تشخیص داده می‌شود و از بین خواهند رفت. میزان تخم‌های رشد کرده مناسب بودن لانه‌های آن منطقه را نشان می‌دهند. هرچه تخم‌های بیشتری در یک ناحیه قادر به زیست باشند و نجات یابند به همان اندازه تمایل بیشتری به آن منطقه اختصاص می‌یابد. بنابراین موقعیتی که در آن بیشترین تعداد تخم‌ها نجات یابند پارامتری خواهد بود که الگوریتم بهینه‌سازی فاخته قصد بهینه‌سازی آن را دارد [15].

فاخته‌ها برای بیشینه کردن نجات تخم‌های خود به دنبال بهترین منطقه می‌گردند. پس از آنکه جوجه‌ها از تخم در آمدند و به فاخته بالغ تبدیل شدند، جوامع و گروه‌هایی تشکیل می‌دهند. هر گروه منطقه سکونت خود را برای زیست دارد. بهترین منطقه سکونت تمام گروه‌ها مقصد بعدی فاخته‌ها در سایر گروه‌ها خواهد بود. تمام گروه‌ها به سمت بهترین منطقه موجود فعلی مهاجرت می‌کنند. هر گروه در منطقه‌ای نزدیک بهترین موقعیت فعلی ساکن می‌شوند. با در نظر گرفتن تعداد تخمی که هر فاخته خواهد گذاشت و همچنین فاصله فاخته‌ها از منطقه بهینه فعلی برای سکونت، تعدادی شعاع تخم‌گذاری محاسبه شده و شکل می‌گیرد. سپس فاخته‌ها شروع به تخم-گذاری تصادفی در لانه‌های داخل شعاع تخم‌گذاری خود می‌کنند. این پروسه تا رسیدن به بهترین محل برای تخم‌گذاری (منطقه با بیشترین سود) ادامه می‌یابد. این محل بهینه جایی است که بیشترین تعداد فاخته‌ها در آن گرد می‌آیند. پس از چند تکرار تمام جمعیت فاخته‌ها به یک نقطه بهینه با حداکثر شباهت تخم‌ها به تخم‌های پرندگان میزبان و همچنین به محل بیشترین منابع غذایی می‌رسند. این محل بیشترین سود کلی را خواهد داشت و در آن کمترین تعداد تخم‌ها از بین خواهند رفت. بنابراین، گام‌های اصلی الگوریتم بهینه‌سازی فاخته را می‌توان به صورت زیر بیان نمود:

1- تشکیل مکان سکونت[1] جمعیت اولیه فاخته‌ها به وسیله‌ی تعدادی نقطه تصادفی
2- تخصیص تعدادی تصادفی تخم به هر فاخته
3- تعریف شعاع تخم گذاری[2] برای هر فاخته
4- هر فاخته تخم خود را در شعاع تخم گذاری مربوط به خود قرار می دهد
5- از بین رفتن تخم‌های فاخته‌ای که توسط پرنده میزبان شناسایی می‌شود
6- اجازه دادن به رشد تخم‌های شناسایی نشده در لانه میزبان
7- ارزیابی محل سکونت فاخته‌های رشد کرده جدید
8- محدود کردن تعداد فاخته‌ها در محیط زندگی با از بین بردن فاخته هایی که در محل‌های بدی زندگی می‌کنند.
9- دسته‌بندی فاخته‌ها و محاسبه بهترین دسته و انتخاب محل سکونت هدف
10- اجاره دادن برای ایجاد جمعیت جدید فاخته‌ها حول محل سکونت هدف
11- اگر شرایط توقف مهیا باشد، الگوریتم متوقف شود وگر نه به مرحله 2 برگردد [15].

7- ایجاد سیگنال مطلوب از مسیر تولیده شده جهت کنترل ربات موازی

پس از اینکه مسیر بهینه ساخته شد و پارامترهای مربوطه بدست آمد، نوبت به ایجاد سیگنال‌های مطلوب جهت کنترل ربات موازی می‌رسد. برای این منظور فرض می‌شود که مجری نهایی ربات موازی نسبت به دستگاه مختصات ثابت O که در شکل 2 نشان داده شده است، با سرعت ثابت خطی V_d مسیر را طی کند، بنابراین با در نظر گرفتن سرعت ثابت در مسیر، معادله (79) بدست می‌آید [18]:

$$\left(\left(\frac{\partial f_x(k(t))}{\partial k(t)}\right)^2+\left(\frac{\partial f_y(k(t))}{\partial k(t)}\right)^2\right)\left(\frac{dk(t)}{d(t)}\right)^2=V_d^2 \qquad k(0)=t_1 \tag{79}$$

در نتیجه، سیگنال‌های مطلوب، جهت تعیین به عنوان موقعیت مرجع ربات به صورت رابطه (80) بیان می‌شوند:

$$x_d=f_x(k(t)) \qquad y_d=f_y(k(t)) \tag{80}$$

با توجه به معادله (79) $k(t)$ به صورت رابطه (81) بدست می‌آید:

$$k(t)=\int_0^t\left(V_d^2\left(\left(\frac{\partial f_x(k(t))}{\partial k(t)}\right)^2+\left(\frac{\partial f_y(k(t))}{\partial k(t)}\right)^2\right)^{-1}\right)^{\frac{1}{2}}dt \tag{81}$$

8- شبیه‌سازی وکنترل ربات صفحه‌ای موازی

جهت شبیه‌سازی ایجاد مسیر بهینه سه محیط با موانع متفاوت در نظر گرفته شده که به ترتیب دارای شرایط پیچیده‌تر و سخت‌تری هستند. در این تحقیق برای تولید مسیر از سه نقطه کنترل در صفحه x و سه نقطه کنترل در صفحه y استفاده شده است. که مجموع این شش نقطه به عنوان متغیرهای بهینه-سازی الگوریتم بهینه‌سازی فاخته در نظر گرفته شده است. اولین محیط، محیطی بدون مانع در نظر گرفته شده که تابع برخورد با مانع در این محیط به صورت رابطه (82) تعریف می‌شود:

$$OB(x,y)=0 \tag{82}$$

نقطه شروع و پایان مسیر به صورتی در نظر گرفته شده اند که ربات موازی توانایی دستیابی به آن‌ها با ابعاد ذکر شده در جدول 1 را داشته باشد.

$$(f_{x_{\text{in}}},f_{x_{\text{end}}})=[0.01 \quad 0.038] \tag{83}$$

$$(f_{y_{\text{in}}},f_{y_{\text{end}}})=[0.01 \quad 0.084] \tag{84}$$

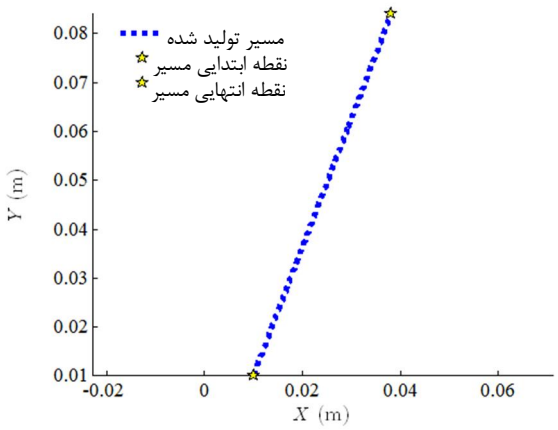
واضح است که کوتاهترین مسیر بهینه، در محیطی که در آن مانعی وجود ندارد قاعدتاً باید خط واصل نقاط شروع و هدف باشد و همانطور که در شکل 7 نشان داده شده است مسیر بهینه ساخته شده تقریباً خط واسط نقاط شروع و هدف است.

1- habitats
2- Egg Laying Radius(ELR)

[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]

جدول 1 مشخصات فیزیکی ربات 3 درجه آزادی

مشخصات فیزیکی یا هندسی	مقدارعددی
طول ضلع($C_iC_{i+1} = b$)	100 mm
جرم سکوی متحرک	0/2 kg
طول($B_iC_i = l$)	200 mm
قطر مقطع بازوی میانی	10 mm
چگالی بازوی میانی	2770 $\frac{kg}{m^3}$
مدول یانگ بازوی میانی	73 GPa
دامنه‌ی کورس(A_iB_i)	200 mm
قطر داخلی سیلندر	14 mm
قطر خارجی سیلندر	16 mm
چگالی سیلندر	2770 $\frac{kg}{m^3}$
طول ضلع(A_iA_{i+1})	660 mm



شکل 7 مسیر تولید شده بدون حضور موانع

دومین محیط، محیطی است که در آن دو دایره به مراکز $x_{c1} = 0.06$ $y_{c1} = 0.06$ $y_{c2} = 0.06$ $x_{c2} = 0.02$ و شعاع‌های $r_{c1} = 0.02$ $r_{c2} = 0.015$ در مسیر ربات موازی قرار گرفته‌اند.

تابع برخورد، نقطه شروع و پایان مسیر به صورت روابط (85) تا (87) درنظر گرفته شده است:

$$OB(x,y) = OB_1 + OB_2 \quad (85)$$

$$(f_{x_{\text{in}}}, f_{x_{\text{end}}}) = [0.01 \quad 0.038] \quad (86)$$

$$(f_{y_{\text{in}}}, f_{y_{\text{end}}}) = [0.01 \quad 0.084] \quad (87)$$

در این محیط انتظار می‌رود که مسیر تولید شده به گونه‌ای باشد که دایره‌ها را دور بزند و از نقاط شروع و هدف تقریبا کوتاه ترین مسیر را طی نماید. همان‌طور که در شکل 8 نشان داده شده است، مسیر تولید شده علاوه بر اینکه به دو مانع دایره‌ای شکل برخوردی ندارد همچنین برای کوتاهتر شدن از نزدیکی یکی از موانع به خوبی گذشته است.

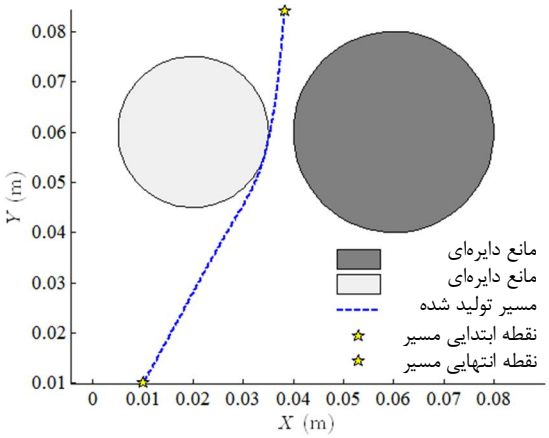
محیط سوم، محیطی است که در آن دو دایره به مراکز $x_{c1} = 0.06$ ، $y_{c1} = 0.06$، $x_{c2} = 0.03$، $y_{c2} = 0$ و شعاع‌های $r_{c1} = 0.02$، $r_{c2} = 0.015$ و یک مستطیل به رئوس $y_{r4} = 0.023$، $x_{r4} = 0.03$، $y_{r1}0.04$، $x_{r1} = -0.015$ در مسیر حرکت ربات موازی قرار گرفته شده است.

تابع برخورد، نقطه شروع و پایان مسیر به صورت روابط (88) تا (90) درنظر گرفته شده‌اند:

$$OB(x,y) = OB_1 + OB_2 + OB_3 \quad (88)$$

$$(f_{x_{\text{in}}}, f_{x_{\text{end}}}) = [0.01 \quad 0.039] \quad (89)$$

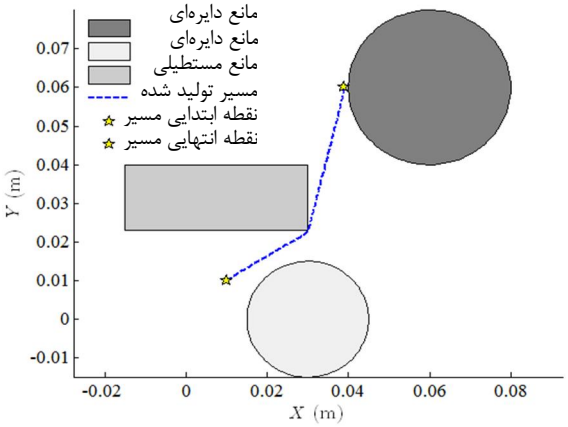
$$(f_{y_{\text{in}}}, f_{y_{\text{end}}}) = [0.01 \quad 0.060] \quad (90)$$



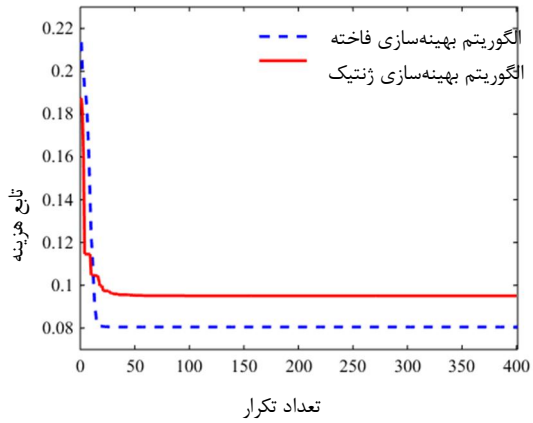
شکل 8 مسیر تولید شده در حضور موانع دایره‌ای

در این محیط انتظار می‌رود که مسیر تولید شده به گونه‌ای باشد که از نزدیکی یکی از رئوس مانع مستطیلی عبور نموده و در ضمن عدم برخورد به دو مانع دایره‌ای شکل، کوتاه‌ترین فاصله را بپیماید تا به نقطه هدف برسد که در شکل 9 نشان داده شده است.

همچنین در شکل 10 مقایسه نرخ همگرایی الگوریتم بهینه سازی فاخته و ژنتیک برای شبیه‌سازی دوم نشان داده شده است. همانطور که در شکل مشخص است الگوریتم بهینه‌سازی فاخته نسبت به الگوریتم ژنتیک تقریباً در تعداد تکرار کمتری به مقدار بهینه‌تر، با سرعت بالاتر رسیده است. سرعت و دقت بالای همگرایی یکی از مزیت‌های این الگوریتم نسبت به روشی چون الگوریتم ژنتیک و یا تجمع ذرات است [15]. همچنین پارامترهای مربوط به الگوریتم‌های فوق در جداول 2 و 3 آورده شده‌اند.



شکل 9 مسیر تولید شده در حضور موانع دایراه‌ای و مستطیلی



شکل 10 مقایسه نرخ همگرایی الگوریتم بهینه‌سازی‌فاخته و ژنتیک در شبیه‌سازی‌دوم

[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]

جدول 2 پارامترهای الگوریتم بهینه‌سازی فاخته

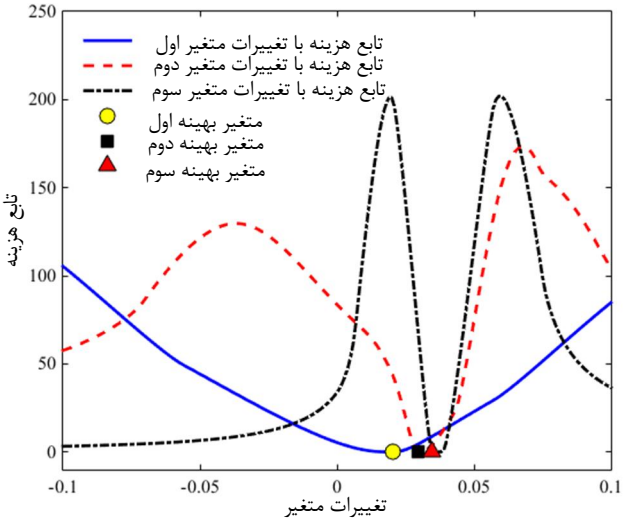
نام پارامتر	مقدار عددی
تعداد فاخته	5
ماکزیمم تعداد فاخته	10
مینیموم تعداد تخم فاخته	10
ماکزیمم تعداد تخم فاخته	15
تعداد تکرار	400
ضریب شعاعی λ	0/5

جدول 3 پارامترهای الگوریتم بهینه‌سازی ژنتیک

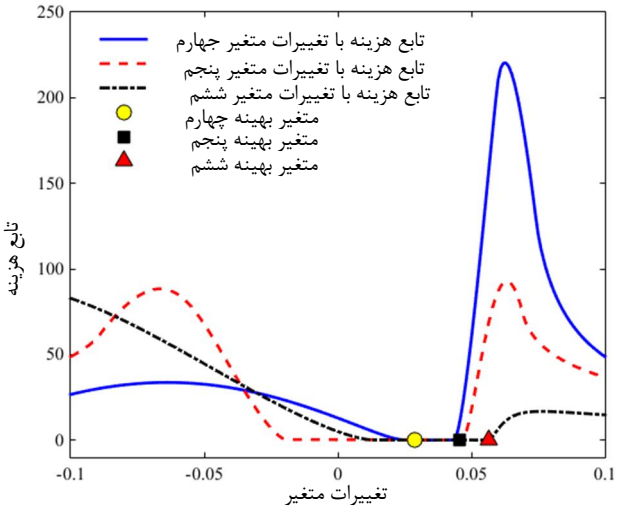
نام پارامتر	مقدار عددی
تعداد جمعیت	200
ترکیب	0/8
جهش	0/04
تعداد تکرار	400

برای اعتبار سنجی نتایج تست همگرایی الگوریتم بهینه‌سازی فاخته به این صورت عمل شد که به تابع هزینه مسیر تولید شده از شش متغیر بهینه بدست آمده، در هر مرتبه پنج متغیر از مقادیر بدست آمده توسط الگوریتم بهینه‌سازی، داده شد و یک متغیر در بازه‌ای متناسب تغییر داده شد. حال باید توابع هزینه‌ای که در این بازه تغییرات بدست می‌آید برای متغیر بهینه داده شده توسط الگوریتم فاخته کمترین مقدار ممکن را داشته باشد. بدین صورت الگوریتم بهینه‌سازی به کمترین مقدار تابع هزینه دست یافته و روش ارائه شده صحه‌گذاری می‌شود. در این تحقیق برای تمام شش متغیر این روند انجام گردید و نتایج حاصل در شکل‌های 11 و 12 ارائه شد. این اعتبار سنجی برای نتایج شبیه‌سازی دوم، انجام گرفته است. فقط ذکر این نکته که به دلیل مقادیر کم تابع هزینه (ناشی از عدم برخورد با موانع) در اطراف متغیر بهینه، در شکل 12 این متغیرها به راحتی قابل تفکیک با سایرین نیستند.

در ادامه تحقیق برای ربات موازی سه درجه آزادی با سه حلقه‌ی سینماتیکی RPR و مشخصات داده شده در جدول 1، کنترلری از نوع مود لغزشی طراحی می‌گردد. ورودی سیستم غیرخطی ربات فوق سه نیروی اعمالی به عملگرهای کشویی در هر سه حلقه سینماتیکی است که از طرف کنترلر طراحی شده ایجاد می‌گردد و خروجی سیستم فوق مکان سکوی متحرک در راستاهای x و y و دوران ϕ است.



شکل 11 نمودار تابع هزینه به ازای تغییرات متغیرهای اول تا سوم



شکل 12 نمودار تابع هزینه به ازای تغییرات متغیرهای چهارم تا ششم

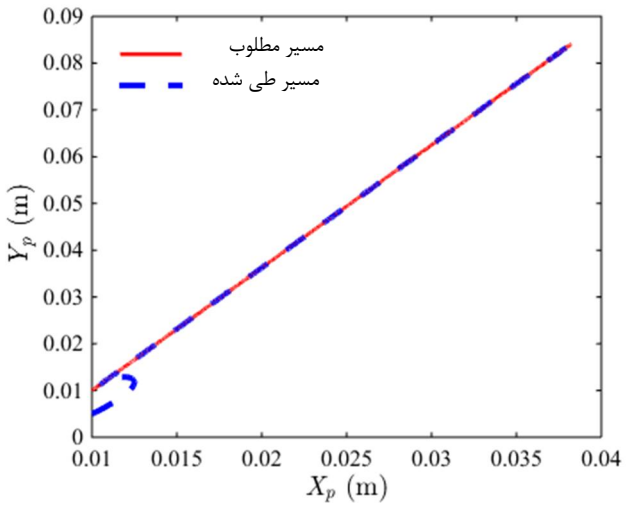
لازم به ذکر است که کلیه ضرایب ثابت مربوط به کنترل کننده مود لغزشی با آزمون و خطا بصورتی بدست آمده است که نیروی اعمالی به هر سه عملگر کشویی ربات موازی قابل اعمال با عملگرهای موجود ساخته شده باشد. این کنترل‌کننده به ازای سه موقعیت مرجع (X_p, Y_p, Φ_p) مورد آزمایش قرار گرفت. به دلیل مقاوم بودن کنترلر به نامعینی‌های پارامتری در این شبیه سازی، مقادیر جرم سکوی متحرک، نیروی خارجی اعمالی به سکوی متحرک در راستای محور x، نیروی خارجی اعمالی به سکوی متحرک در راستای محور y و اینرسی مجری نهایی ربات موازی متغیر و در بازه‌ای مجاز به شرح جدول 4 تغییرات دارند.

مجری نهایی ربات موازی در تمام شبیه‌سازی‌ها در مکان اولیه‌ی $[0.01\ \mathrm{m}, 0.005\ \mathrm{m}, 0.05\ \mathrm{rad}]$ قرار گرفته است و مجری نهایی ربات در تمام شبیه‌سازی‌ها مسیر را با سرعت ثابت $V_d = 1\frac{\mathrm{cm}}{\mathrm{s}}$ طی می‌نماید. در این بخش سه مسیر تولید شده به عنوان موقعیت مرجع برای ربات موازی تعیین شده است. همچنین زاویه مجری نهایی که مستقل از تولید مسیر فوق است در هر سه شبیه‌سازی به ترتیب $\phi_p = 0.06, 0.08, 0.1\ \mathrm{Rad}$ در نظر گرفته شده است. همان‌طور که در شکل‌های 13 تا 18 مشخص است ربات بعد از طی مدت زمان اندکی به موقعیت مرجع تعریف شده $(X_{pd}, Y_{pd}, \Phi_{pd})$ دست یافته است. همچنین شکل‌های 19 تا 21 مقدار نیروی اعمالی به عملگرهای ربات سه درجه آزادی موازی از طرف کنترل‌کننده طراحی شده در طی این مدت زمان را نشان می‌دهد که اندازه آن‌ها مطلوب بوده و قابل ایجاد توسط عملگرهای کشویی ساخته شده موجود است.

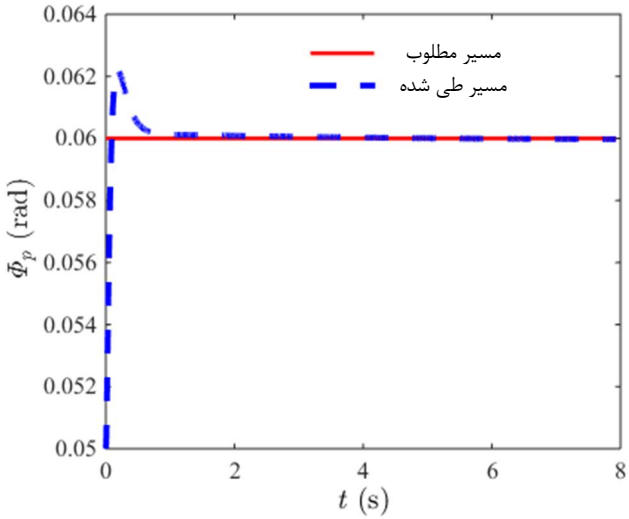
جدول 4 نامعینی‌های پارمتری ربات 3 درجه آزادی

نام پارمتر	مقدار اسمی در معادلات کنترلر	مقدار تعیین شده در شبیه سازی
جرم سکوی متحرک	kg 0/2	kg 0/5
نیروی خارجی اعمالی به سکوی متحرک در راستای محور x	N 0	N 10
نیروی خارجی اعمالی به سکوی متحرک در راستای محور y	N 0	N 2
اینرسی مجری نهایی I_e	$\mathrm{kg \cdot m^2}$ -5e 8/33	$\mathrm{kg \cdot m^2}$ -4e2 /083

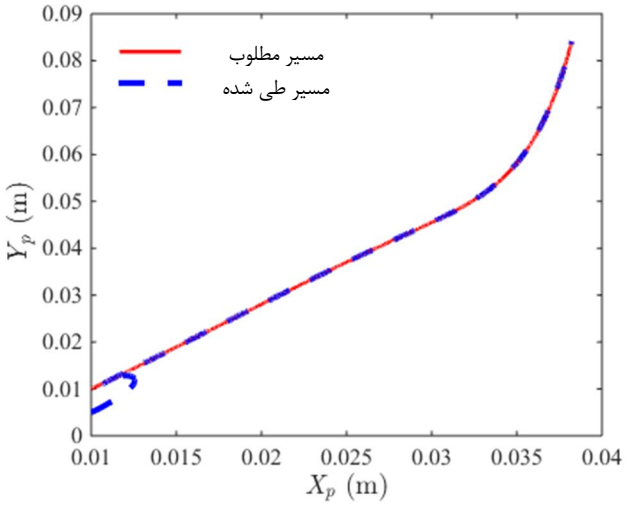
[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]



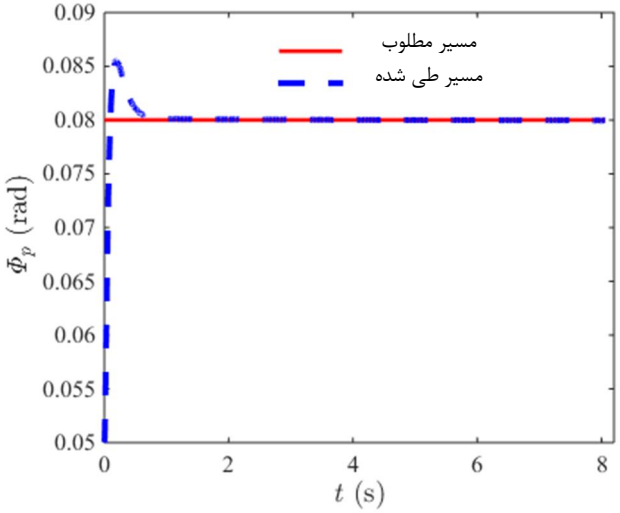
شکل 13 مکان مجری نهایی ربات در شبیه‌سازی اول



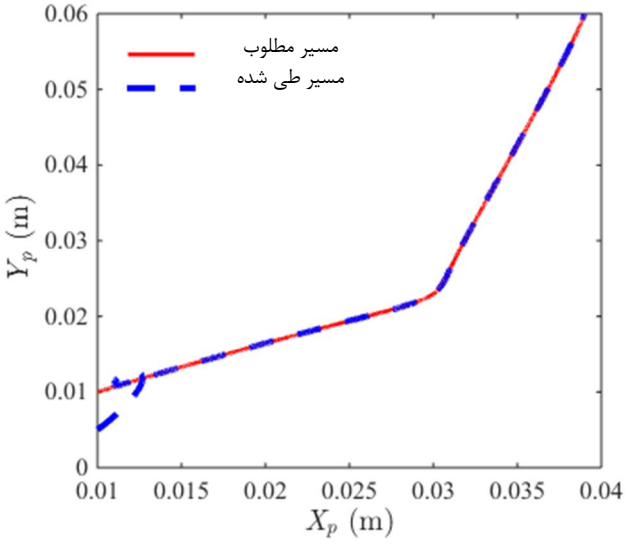
شکل 14 چرخش مجری نهایی ربات در شبیه‌سازی اول



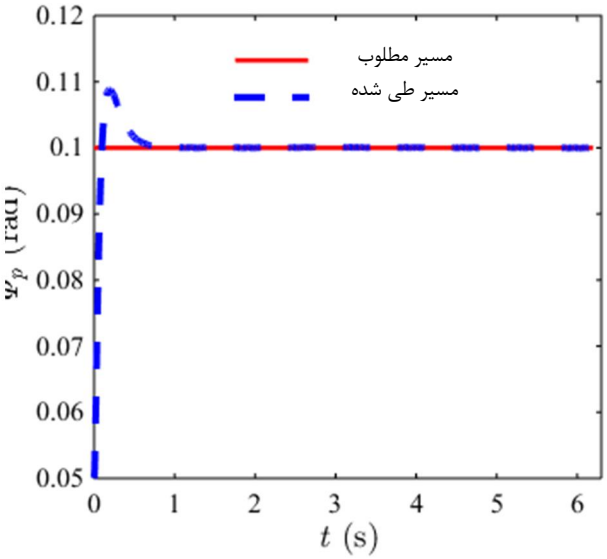
شکل 15 مکان مجری نهایی ربات در شبیه‌سازی دوم



شکل 16 چرخش مجری نهایی ربات در شبیه‌سازی دوم

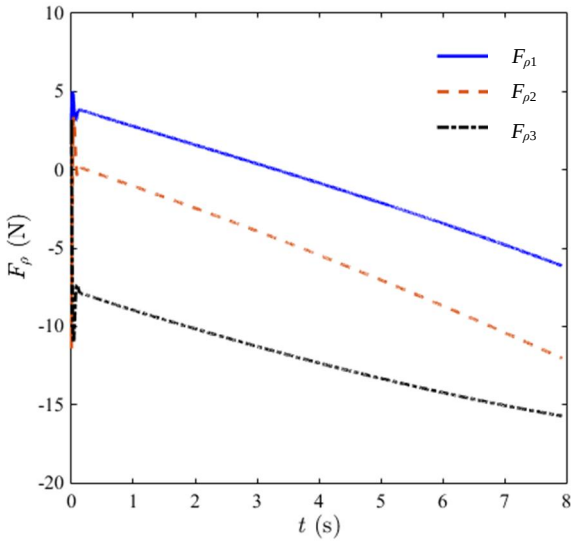


شکل 17 مکان مجری نهایی ربات در شبیه‌سازی سوم

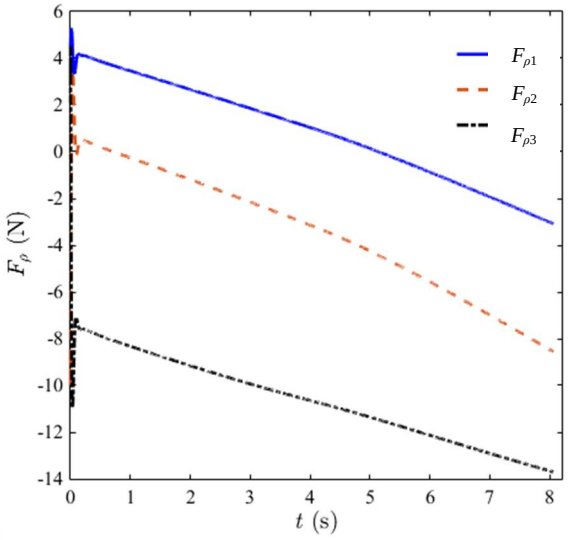


شکل 18 چرخش مجری نهایی ربات در شبیه‌سازی سوم

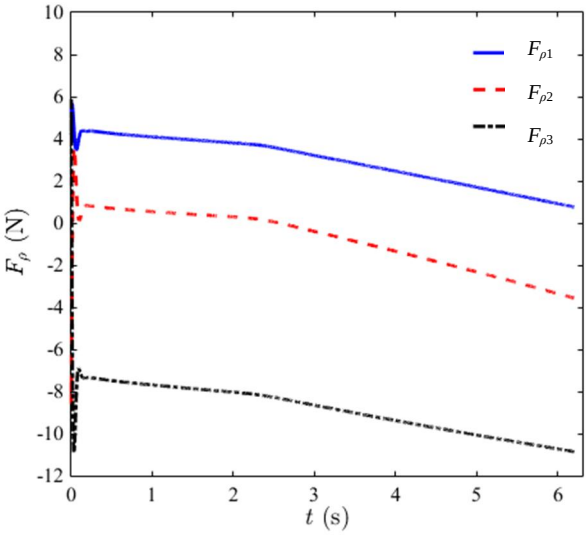
[Downloaded from mme.modares.ac.ir on 2024-10-12]
[DOR: 20.1001.1.10275940.1394.15.2.32.0]



شکل 19 نیروی اعمال شده به عملگرهای کشویی در شبیه‌سازی اول



شکل 20 نیروی اعمال شده به عملگرهای کشویی در شبیه‌سازی دوم



شکل 21 نیروی اعمال شده به عملگرهای کشویی در شبیه‌سازی سوم

9- نتیجه‌گیری

در تحقیق پیش‌رو برای یک ربات سه درجه آزادی صفحه‌ای موازی مسیرهایی با وجود موانع و به وسیله‌ی منحنی اسپلاین و با استفاده از الگوریتم کارامد فاخته طراحی گردید. همان‌طور که از نتایج ارائه شده در این تحقیق مشخص است استفاده از روش منحنی‌های اسپلاین و الگوریتم بهینه-سازی فاخته به خوبی مؤثر واقع شده به صورتی که طول مسیر ایجاد شده تا حد امکان، بدون برخورد با موانع تعریف شده کوتاه شده است. برای مقایسه سرعت و دقت همگرایی الگوریتم فوق، نتایج همگرایی این الگوریتم با الگوریتم مشهور ژنتیک مقایسه گردید که بررسی نتایج نشان از سرعت و دقت بالای آن دارد. همچنین نتایج همگرایی این الگوریتم توسط روشی صحه‌گذاری شد. در ادامه مسیرهای تعریف شده با استفاده از روشی به عنوان سیگنال مطلوب برای موقعیت مرجع سکوی متحرک ربات تعریف شدند و ربات با استفاده از کنترل‌کننده مود لغزشی با وجود نامعینی‌های زیاد تعریف شده توانست به آن موقعیت‌ها با طی مدت زمان اندکی دست یابد. همچنین از نتایج مربوط به نیروهای وارد شده به عملگرهای کشویی مشخص است، نیروهای اعمال شده قابلیت ایجاد توسط عملگرهای ساخته شده موجود را دارند.

10- مراجع

[1] M. Terrier, A. Dugas and J.Y. Hascoet, Qualification of parallel kinematics machines in high-speed milling on free form surfaces, *International Journal of Machine Tools and Manufacture,* Vol. 44, pp. 865-877, 2004.

[2] L. Ren, J. K. Mills, and D. Sun, Adaptive Synchronization Control of a Planar Parallel Manipulator, *Proceeding of The American Control Conference,* pp. 1–6, 2004.

[3] H.Kordjazi, A.Akbarzadeh, Control of 3-PRR parallel robots using computed torque method, *Tenth Conference on Manufacturing Engineering,* Babol, Iran, 2010. (In Persian)

[4] S.N.Nabavi, A.Akbarzadeh, S.Abolghasemi, A Method for Solving Dynamic Equations of a 3-PRR Parallel Robot, *A. Mechanics,* vol. 232, pp. 414–418, 2012.

[5] X. Zhang, J. K. Mills, and W. L. Cleghorn, Multi-mode vibration control and position error analysis of parallel manipulator with multiple flexible links, *Transactions of the Canadian Society for Mechanical Engineering,* vol. 34, no. 2, pp. 197–213, 2010.

[6] X. Zhang, J. K. Mills, and W. L. Cleghorn, Dynamic Modeling and Experimental Validation of a 3-PRR Parallel Manipulator with Flexible Intermediate Links, *J. Intell.Robot.Syst,* vol. 50, no. 4, pp. 323–340, 2007.

[7] G. Javid, M.-R.Akbarzadeh-T., A. Akbarzadeh, and S. N. Nabavi, Trajectory tracking of 3-PRR parallel manipulator with PI Adaptive fuzzy terminal sliding mode controller, *1st International conference on Computer and Knowledge Engineering (ICCKE),* Mashhad, Iran, 2011.

[8] A. Eshaghiyeh Firoozabadi, S.Ebrahimi and G.Amirian, Dynamic characteristics of a 3-RPR planar parallel manipulator with flexible intermediate links, *Robotica,* 2014 (Published online)

[9] R. W. II and A. Joshi, Planar parallel 3-RPR manipulator, *Proceedings of the Sixth Conference on Applied Mechanisms and Robotics,* 1999.

[10] S. S. Parsa, H. M. Daniali, and R. Ghaderi, Optimization of parallel manipulator trajectory for obstacle and singularity avoidances based on neural network, *Int J Adv Manuf Technol,* pp. 811–816, 2010.

[11] M. Danesh, M.R Abazari, Optimize path planning of a planar parallel robot to avoid obstacle collision using genetic algorithm, *Intelligent Systems (ICIS), 2014 Iranian Conference on,* vol., no., pp.1,4, 4-6 Feb. 2014.

[12] C. Nasa, S. Bandyopadhyay, Trajectory-tracking control of a planar 3-RRR parallel manipulator with singularity avoidance, *13th World Congress in Mechanism and Machine Science,* Guanajuato, Mexico, 19-25 June, 2011.

[13] S. Lahouar, S. Zeghloul, L. Romdhane, Singularity Free Path Planning for Parallel Robots, Advances in Robot Kinematics: Analysis and Design, 2008

[14] De Boor, C., *A Practical Guide to Splines, Springer-Verlag,* 1978.

[15] R. Rajabioun, Cuckoo Optimization Algorithm, *Applied Soft Computing,* 2011.

[16] J. J. E. Slotine and W. Li, *Applied nonlinear control, Prentice hall Englewood Cliffs,* 1991.

[Downloaded from mme.modares.ac.ir on 2024-10-12]

[DOR: 20.1001.1.10275940.1394.15.2.32.0]

[17] F. Piltan, S. Emamzadeh, and M. Mirzaei, PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB / SIMULINK and Their Integration into Graduate / Undergraduate Nonlinear Control, Robotics and MATLAB Courses, *IJRA,* 2012.

[18] E. Zakeri and S. Farahat, Safe path planning and control of an Unmanned Underwater Vehicle (UUV) using particle swarm optimization and fuzzy logic control method*, Moadres Mechanical Engineering* , vol. 14, no. 14, pp. 199–210, 2015. (In Persian)

[Downloaded from mme.modares.ac.ir on 2024-10-12]

[DOR: 20.1001.1.10275940.1394.15.2.32.0]