مجله مهندسی مکانیک مدرس، دی 1396، دوره 17، شماره 10، ص‌ص 20-28

ماهنامه علمی پژوهشی

مهندسی مکانیک مدرس

mme.modares.ac.ir

# طراحی کنترل مدل پیش‌بین با استفاده از توابع لاگر و ضرایب وزنی نمایی برای ربات متحرک چرخ‌دار

**مجید شهباززاده[1]، سید جلیل ساداتی رستمی[2*]، سارا میناگر[3]**

1- دانشجوی کارشناسی‌ارشد، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

2- استادیار، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

3- استادیار، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

* بابل، صندوق‌پستی 71167-47148، j.sadati@nit.ac.ir

**اطلاعات مقاله**

مقاله پژوهشی کامل
دریافت: 17 تیر 1396
پذیرش: 06 شهریور 1396
ارائه در سایت: 07 مهر 1396

*کلید واژگان:*
ربات متحرک چرخ‌دار
قید غیرهولونومیک
کنترل مدل پیش‌بین
ردیابی مسیر
ضرایب وزنی نمایی

**چکیده**

در سال‌های اخیر مطالعات زیادی بر کنترل حرکت ربات‌های چرخ‌دار شکل گرفته است. در این میان مسأله ردیابی مسیر یکی از مطرح‌ترین این مسائل بوده است. یکی از روش‌های دور زدن محدودیت‌های قضیه براکت به منظور ردیابی مسیر این است که از ساختار کنترلی پیشخور و پسخور استفاده کنیم. کنترل پیشخور از مسیر مرجع محاسبه می‌شود و به تنهایی کارا نیست، چون قابلیت جبران خطاهای اندازه‌گیری و خطاهای حالت اولیه را ندارد؛ بنابراین بخش کنترلی پسخور نیز مورد نیاز است. در این مقاله از روش کنترل مدل پیش‌بین به عنوان بخش کنترلی پسخور استفاده کرده‌ایم. با توجه به این‌که شرایط اولیه سیستم در بیشتر مواقع بر مسیر مطلوب قرار ندارد، ردیابی هرچه سریع‌تر مسیر مرجع در زمان‌های اولیه اهمیت ویژه‌ای دارد. از این‌رو در این مقاله قصد داریم خطای ردیابی مسیر مرجع در زمان‌های اولیه را با استفاده از توابع لاگر در کنترل مدل پیش‌بین و همچنین کنترل مدل پیش‌بین با ضرایب وزنی نمایی کاهش دهیم. نتایج شبیه‌سازی نشان می‌دهد که ردیابی مسیر مرجع با استفاده از توابع لاگر در کنترل مدل پیش‌بین نسبت به حالت‌هایی که از کنترل مدل پیش‌بین با استفاده از ضرایب وزنی نمایی و همچنین کنترل مدل پیش‌بین معمولی استفاده شده بهبود یافته است.

# Designing Model Predictive Control for Trajectory Tracking of a Wheeled Mobile Robot Using Laguerre functions and Exponential Data Weighting

**Majid Shahbazzadeh[1], Seyed Jalil Sadati Rostami[2*], Sara Minagar[3]**

Department of Electrical and computer Engineering, Babol Noshirvani University of Technology, Babol, Iran
* P.O.B. 47148-71167, Babol, Iran, j.sadati@nit.ac.ir

**ARTICLE INFORMATION**

Original Research Paper
Received 08 July 2017
Accepted 28 August 2017
Available Online 29 September 2017

*Keywords:*
Wheeled mobile robot
Nonholonomic constraint
Model predictive control
Trajectory tracking
Exponential data weighting

**ABSTRACT**

Numerous studies have been devoted to motion control of wheeled mobile robots in recent years. Among them, trajectory tracking has received much attention. A feed-forward and feedback control structure for trajectory tracking is used to circumvent the limitation of Brockett's theorem. Feed-forward control is calculated according to the reference trajectory, it cannot compensate instrumentation and initial state errors, and therefore a feedback controller is utilized as well. In this paper a model predictive controller is used as the feedback controller. Since the initial state is not often matched to the desired trajectory, rapid tracking of the trajectory in early steps is very important. In this paper a model predictive controller with Laguerre functions and a model predictive controller with exponential data weighting is used to reduce tracking error in early steps. According to simulation results, reference trajectory tracking is improved through Laguerre functions in model predictive controller.

## 1- مقدمه

ربات یکی از مهم‌ترین ساخته‌های دست بشر است. ربات‌ها بخش مهمی از زندگی مدرن بشر امروزی را به خود اختصاص داده‌اند [1]. صنایع نظامی، صنایع پزشکی و کارخانه‌های بزرگ با سرعت هرچه بیشتری به سمت مکانیزه شدن پیش می‌روند. قدرت مانور بالا، متحرک بودن و فضای کاری وسیع‌تر مزایایی است که ربات‌های متحرک[1] نسبت به ربات‌های ثابت دارند [2]. همین مزایا باعث شده است که ربات‌های متحرک کاربردهای فراوانی داشته باشند. این ربات‌ها نه تنها برای کار در کارخانه‌ها، بلکه برای کارهای دیگری که برای انسان مشکل است و یا در موقعیت‌هایی استفاده می‌شود که امکان دارد برای انسان خطرناک باشد. منظور از موقعیت‌های خطرناک فقط جنگ و ماموریت‌های نظامی نیست، بلکه می‌توان به مواردی مانند کشف مواد معدنی در کوه‌ها، کشف نفت و گاز در زیر زمین و یا پالایشگاه‌ها اشاره کرد [4,3].

کنترل حرکت ربات‌های متحرک چرخ‌دار علاوه‌بر کاربردهای عملی مهم، مسأله تحقیقاتی مهمی هم به شمار می‌آید. از اوایل دهه 1980 میلادی مطالعات زیادی بر کنترل ربات‌های متحرک شکل گرفت. در این میان مسأله دنبال کردن مسیر بسیار مورد توجه محققین قرار گرفته است. از این‌رو قوانین کنترلی متفاوتی برای حرکت ربات‌های چرخ‌دار با سینماتیک دیفرانسیلی ارائه

[1] Mobile robots

**Please cite this article using:**

**برای ارجاع به این مقاله از عبارت ذیل استفاده نمایید:**

M. Shahbazzadeh, S. J. Sadati Rostami, S. Minagar, Designing Model Predictive Control for Trajectory Tracking of a Wheeled Mobile Robot Using Laguerre functions and Exponential Data Weighting, *Modares Mechanical Engineering*, Vol. 17, No. 10, pp. 20-28, 2017 (in Persian)

شده است [5]. براساس قضیه براکت[1] هیچ فیدبک نامتغیر با زمان همواری از متغیرهای حالت نمی‌توان یافت که سیستم غیرهولونومیک[2] را حول نقطه تعادل به طور مجانبی پایدار کند [6].

یکی از روش‌های کنترل ربات متحرک چرخ‌دار طراحی کنترل‌کننده غیرخطی است. نخستین بار در [7-9] قانون کنترل غیرخطی براساس تئوری لیاپانوف طراحی شده است. پس از آن محققین بسیاری بر ساختار کنترل‌کننده غیرخطی ربات متحرک چرخ‌دار تحقیق کرده‌اند، مانند به‌کارگیری کنترل تطبیقی [10]، تعمیم آن به فازی [11]، خطی‌سازی ورودی- خروجی [12]، کنترل‌کننده ترکیبی و طراحی رویت‌گر در [13] و بسیاری دیگر در [14] قانون کنترل غیرخطی براساس تئوری لیاپانوف مشکل ناپیوستگی در خطای جهت‌گیری را رفع می‌کند.

می‌توان برای کنترل بسیاری از ربات‌های تجاری فقط مدل سینماتیکی را در نظر گرفت، زیرا کنترل‌کننده داخلی از پیش برای آن طراحی شده است که دینامیک ربات را مدیریت می‌کند. در غیر این صورت باید یک جبران‌ساز دینامیکی پیش از اعمال کنترل‌کننده سینماتیکی برای سیستم پیاده‌سازی کرد [15].

کنترل مدل پیش‌بین یکی از روش‌هایی است که می‌تواند به ربات متحرک چرخ‌دار اعمال شود تا مسیر مرجع از پیش تعیین شده‌ای را دنبال کند. روش کنترل مدل پیش‌بین معمولاً با مدل سینماتیکی و یا دینامیکی خطی شده سروکار دارد. تعدادی از روش‌های کنترل مدل پیش‌بین ربات متحرک چرخ‌دار در سال‌های اخیر گزارش شده است. مانند کنترل مدل پیش‌بین تعمیم یافته در [16]، کنترل مدل پیش‌بین با پیش‌بینی‌کننده اسمیت[3] برای مدیریت کردن تأخیر زمانی سیستم [17]، کنترل مدل پیش‌بین براساس مدل خطی متغیر با زمان در [18]، کنترل‌کننده مدل پیش‌بین غیرخطی با شبکه عصبی چندلایه در [19] توصیف شده است. در بسیاری از روش‌ها قانون کنترل از بهینه‌سازی تابع هزینه به دست می‌آید. یک روش برای به دست آوردن قانون کنترل حل تحلیلی آن است. از این‌رو محاسبات کمتر و سرعت پیاده‌سازی آن بیشتر می‌شود [20].

در این تحقیق برای کنترل ربات متحرک چرخ‌دار از ساختار کنترلی پیشخور و پسخور استفاده شده است. برای محاسبه کنترل پسخور از کنترل‌کننده مدل پیش‌بین با حل تحلیلی آن استفاده شده است، همچنین تکنیک وزن‌دار کردن نمایی داده‌ها جهت بهبود ردیابی مسیر مرجع به کار رفته است.

ساختار ادامه این تحقیق به شرح زیر است: در بخش 2 در مورد محدودیت‌های سینماتیکی به طور مختصر توضیح داده شده و سپس به بحث در مورد مدل سینماتیکی ربات متحرک چرخ‌دار دیفرانسیلی پرداخته شده است. در بخش 3 به منظور کنترل ردیابی مسیر به ساختار کنترلی پیشخور و پسخور پرداخته که برای محاسبه کنترل پسخور از کنترل‌کننده مدل پیش‌بین استفاده شده است، سپس به توصیف کنترل مدل پیش‌بین با ضرایب وزنی نمایی و کنترل مدل پیش‌بین با توابع لاگر به منظور بهبود ردیابی مسیر مرجع پرداخته شده است. در بخش 4 کنترل‌کننده‌های به دست آمده بر ربات متحرک چرخ‌دار شبیه‌سازی شده و نتایج آن با کنترل‌کننده مدل پیش‌بین معمولی مقایسه شده است. در بخش 5 نتیجه‌گیری مقاله ارائه شده است.

[1] Brockett
[2] Nonholonomic
[3] Smith-predictor

## 2- مدل سینماتیکی و محدودیت‌های آن

محدودیت‌های سینماتیکی از ساختار ربات و مدل سینماتیکی آن نشأت می‌گیرند. این محدودیت‌ها می‌توانند هولونومیک[4] یا غیرهولونومیک باشند [5].

### 2-1- قیدهای هولونومیک

این قیدها به متغیرهای حالت توصیف شده سیستم بستگی دارند. برای یک سیستم با $n$ متغیر حالت $q(t) = [q_1, \ldots, q_n]^T$، قید هولونومیک به صورت رابطه (1) بیان می‌شود.

$$f(q) = f(q_1, \ldots, q_n) = 0 \tag{1}$$

به طوری که $f$ و مشتقاتش تابع‌هایی پیوسته هستند. قید رابطه (1) می‌تواند برخی متغیرهای حالت خاص را حذف کند. وقتی $m$ قید هولونومیک مستقل از زمان خطی داشته باشیم، مختصات تعمیم یافته زیرفضایی با بعد $(n-m)$ را تشکیل می‌دهد.

### 2-2- قیدهای غیرهولونومیک

قیدهای غیرهولونومیک سرعت‌های ممکن سیستم یا به عبارتی مسیرهای حرکتی ممکن را محدود می‌کنند که می‌توان به صورت رابطه (2) نوشت [5].

$$f(q, \dot{q}) = f(q_1, \ldots, q_n, \dot{q}_1, \ldots, \dot{q}_n) = 0 \tag{2}$$

به طوری که $f$ تابعی هموار با مشتقات پیوسته و $\dot{q}$ برداری از سرعت‌های سیستم در مختصات تعمیم یافته است. اگر سیستم قید رابطه(2) را نداشته باشد محدودیتی در مسیرهای حرکتی ندارد.

قید سینماتیکی رابطه (2) اگر انتگرال‌پذیر باشد، هولونومیک است، به این معنی که سرعت‌ها $\dot{q}_1, \ldots, \dot{q}_n$ می‌توانند از معادله رابطه (2) حذف شوند و قید به فرم رابطه (1) به‌دست آید.

اگر قید رابطه (2) انتگرال‌پذیر نباشد غیرهولونومیک است. قیدهای غیرهولونومیک فضای دسترسی به سرعت‌های سیستم را کاهش می‌دهند. اگر $m$ قید غیرهولونومیک مستقل خطی به صورت رابطه (2) داشته باشیم، بعد فضای سرعت آن $(n-m)$ می‌شود.

### 2-3- مدل سینماتیکی

مدل سینماتیکی رابطه بین ورودی‌های سیستم و رفتار آن بر فضای حالت سیستم را تشریح می‌کند. در ربات‌هایی که با سرعت کمی حرکت می‌کنند، دینامیک ربات به ندرت مورد توجه قرار می‌گیرد و مسأله مسیریابی به شکل ساده‌تری در نظر گرفته می‌شود. استفاده از مدل سینماتیکی برای ربات‌هایی که با سرعت بالایی حرکت می‌کنند پاسخ مناسبی ارائه نمی‌دهد [21].

موقعیت ربات در صفحه توسط برداری از متغیرهای حالتش به صورت رابطه (3) مشخص می‌شود.

$$q(t) = [x(t), y(t), \varphi(t)]^T \tag{3}$$

چارچوب مختصات جهانی $(X_g, Y_g)$ در شکل 1 نشان داده شده است. $(X_m, Y_m)$ چارچوب متحرک ربات است. رابطه بین چارچوب جهانی و متحرک توسط ماتریس رابطه (4) به دست می‌آید.

$$R(\varphi) = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

ربات متحرک چرخ‌دار دیفرانسیلی مکانیزم حرکتی بسیار ساده‌ای دارد.

[4] Holonomic

سرعت هر چرخ توسط یک موتور مجزا کنترل می‌شود. نقطه مشترک تقاطع محورهای چرخ‌ها مرکز چرخش آنی[1] تعریف می‌شود. متغیرهای دیگر به صورت زیر بیان می‌شود: $r$ شعاع چرخ، $L$ فاصله بین چرخ‌ها و $R(t)$ شعاع آنی مسیر است. مطابق شکل 1 در هر لحظه‌ای از زمان چرخ‌ها سرعت زاویه‌ای یکسانی به صورت رابطه (5) حول مرکز چرخش آنی دارند.

$$\omega = \frac{v_{\mathrm{L}}(t)}{R(t)-\frac{L}{2}} = \frac{v_{\mathrm{R}}(t)}{R(t)+\frac{L}{2}} \tag{5}$$

$\omega(t)$ و $R(t)$ به صورت روابط (7,6) به‌دست می‌آیند.

$$\omega(t) = \frac{v_{\mathrm{R}}(t)-v_{\mathrm{L}}(t)}{L} \tag{6}$$

$$R(t) = \frac{L}{2}\frac{v_{\mathrm{R}}(t)+v_{\mathrm{L}}(t)}{v_{\mathrm{R}}(t)-v_{\mathrm{L}}(t)} \tag{7}$$

سرعت خطی به صورت رابطه (8) محاسبه می‌شود

$$v(t) = \omega(t)R(t) = \frac{v_{\mathrm{R}}(t)+v_{\mathrm{L}}(t)}{2} \tag{8}$$

سرعت‌های خطی چرخ‌های چپ و راست به ترتیب $v_{\mathrm{L}}(t) = r\omega_{\mathrm{L}}(t)$ و $v_{\mathrm{R}}(t) = r\omega_{\mathrm{R}}(t)$ و سرعت‌های زاویه‌ای چرخ‌های چپ و راست به ترتیب حول محورشان $\omega_{\mathrm{L}}(t)$ و $\omega_{\mathrm{R}}(t)$ است. سینماتیک داخلی ربات (در مختصات محلی) را می‌توان به صورت رابطه (9) بیان کرد.

$$\begin{bmatrix}\dot{x}_{\mathrm{m}}\\ \dot{y}_{\mathrm{m}}\\ \dot{\varphi}\end{bmatrix} = \begin{bmatrix}vX_{\mathrm{m}}(t)\\ vY_{\mathrm{m}}(t)\\ \omega(t)\end{bmatrix} = \begin{bmatrix}\frac{r}{2} & \frac{r}{2}\\ 0 & 0\\ -\frac{r}{L} & \frac{r}{L}\end{bmatrix}\begin{bmatrix}\omega_{\mathrm{L}}(t)\\ \omega_{\mathrm{R}}(t)\end{bmatrix} \tag{9}$$

و سینماتیک خارجی ربات (در مختصات جهانی) به صورت رابطه (10) به‌دست می‌آید.

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\varphi}\end{bmatrix} = \begin{bmatrix}\cos(\varphi(t)) & 0\\ \sin(\varphi(t)) & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v(t)\\ \omega(t)\end{bmatrix} \tag{10}$$

به طوری که $v$ و $\omega$ ورودی‌های کنترل هستند. مدل رابطه (10) را می‌توان با تقریب اویلر به فرم گسسته زیر نوشت.

$t = kT_{\mathrm{s}}, k = 0,1,2,\dots$ و $T_{\mathrm{s}}$ زمان نمونه برداری سیستم به صورت رابطه (11) است.

$$\begin{aligned} x(k+1) &= x(k) + v(k)T_{\mathrm{s}}\cos(\varphi(k))\\ y(k+1) &= y(k) + v(k)T_{\mathrm{s}}\sin(\varphi(k))\\ \varphi(k+1) &= \varphi(k) + \omega(k)T_{\mathrm{s}} \end{aligned} \tag{11}$$

## 3- کنترل ردیابی مسیر

کنترل ردیابی مسیر مکانیزمی است که باعث می‌شود حرکت ربات تا حد امکان به مسیر مرجع نزدیک شود. برای کنترل ردیابی مسیر موفق نیاز به استفاده از کنترل‌کننده ناهموار یا متغیر با زمان است، زیرا سیستم غیرخطی و متغیر با زمان است. هنگامی که ربات حرکت می‌کند باید قیدهای غیرهولونومیک در نظر گرفته شوند، چون مسیر آن نمی‌تواند دلخواه باشد. یک دلیل استفاده از روش‌های کنترل ردیابی مسیر واقعیتی است که ربات‌ها معمولاً در محیط‌هایی با حضور موانع یا محدودیت‌های متفاوت حضور دارند و تعریف مسیر مرجع سبب می‌شود که ربات به سمت مسیر مطلوب هدایت شود.

[1] Instantaneous Center of Rotation (ICR)

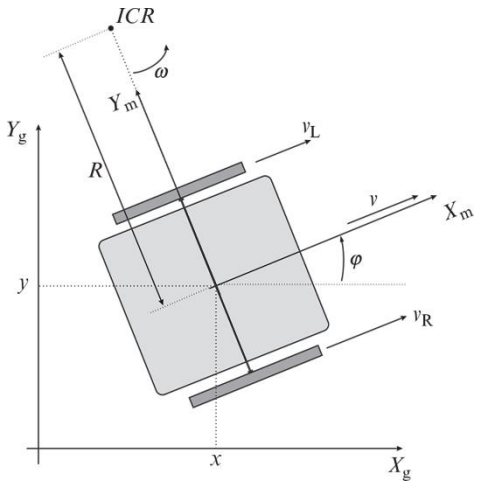


**Fig. 1** Differential drive kinematics

**شکل 1** سینماتیک رانش دیفرانسیلی

یکی از روش‌های دور زدن محدودیت‌های قضیه براکت این است که از ساختار کنترلی پیشخور[2] و پسخور[3] را استفاده کنیم.

### 3-1- کنترل پیشخور

سیستم ربات متحرک چرخ‌دار مسطح[4] دیفرانسیلی است، یعنی تمام متغیرهای حالت و ورودی‌های سیستم به عنوان تابع‌هایی از خروجی‌های مسطح و تعداد محدودی از مشتقات زمانی آن‌ها می‌توانند بازنویسی شوند. از این‌رو تابع‌های $\mathcal{F}_x$ و $\mathcal{F}_u$ به صورت رابطه (12) تعریف می‌شوند.

$$\begin{aligned} X &= \mathcal{F}_x\left(\xi, \dot{\xi}, \ddot{\xi}, \dots, \frac{\mathrm{d}^p}{\mathrm{d}t^p}\xi\right)\\ u &= \mathcal{F}_u\left(\xi, \dot{\xi}, \ddot{\xi}, \dots, \frac{\mathrm{d}^p}{\mathrm{d}t^p}\xi\right) \end{aligned} \tag{12}$$

در آن بردارهای $X$، $u$ و $\xi$ به ترتیب حالت‌های سیستم، ورودی آن‌ها و خروجی‌های مسطح و $p$ یک عدد محدود است.

در سینماتیک سیستم دیفرانسیلی رابطه (10) خروجی‌های مسطح همان $x$ و $y$ است. به آسانی می‌توان نشان داد که ورودی‌ها و متغیر حالت سوم (جهت‌گیری ربات) را می‌توان به صورت تابع‌هایی از $x$ و $y$ و مشتقات آن‌ها نشان داد.

می‌دانیم که $\dot{x}$ و $\dot{y}$ سرعت‌های خطی ربات در مختصات کارتزین است؛ بنابراین سرعت خطی به صورت رابطه (13) به‌دست می‌آید.

$$v(t) = \sqrt{\dot{x}^2(t) + \dot{y}^2(t)} \tag{13}$$

ربات متحرک چرخ‌دار دیفرانسیلی به دلیل وجود قیدهای غیرهولونومیک در سمت جهت‌گیریش حرکت می‌کند؛ بنابراین جهت‌گیری ربات به صورت رابطه (14) محاسبه می‌شود.

$$\varphi(t) = \arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right) \tag{14}$$

در نهایت با مشتق گرفتن $\varphi$ برحسب زمان سرعت زاویه‌ای $\omega(t)$ به صورت رابطه (15) به‌دست می‌آید.

$$\omega(t) = \frac{\mathrm{d}}{\mathrm{d}t}\left[\arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right)\right] = \frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{\dot{x}^2(t) + \dot{y}^2(t)} \tag{15}$$

[2] Feedforward
[3] Feedback
[4] Flat

در ربات‌های متحرک چرخ‌دار دیفرانسیلی سرعت‌های مرجع $v_{\mathrm{ref}}(t)$ و $\omega_{\mathrm{ref}}(t)$ از مسیرمرجع توسط $x_{\mathrm{ref}}(t)$ و $y_{\mathrm{ref}}(t)$، براساس روابط (15,13) به صورت روابط (17,16) به‌دست آورده می‌شوند.

$$v_{\mathrm{ref}}(t) = \sqrt{\dot{x}^2{}_{\mathrm{ref}}(t) + \dot{y}^2{}_{\mathrm{ref}}(t)} \tag{16}$$

$$\omega_{\mathrm{ref}}(t) = \frac{\dot{x}_{\mathrm{ref}}(t)\ddot{y}_{\mathrm{ref}}(t) - \dot{y}_{\mathrm{ref}}(t)\ddot{x}_{\mathrm{ref}}(t)}{\dot{x}^2{}_{\mathrm{ref}}(t) + \dot{y}^2{}_{\mathrm{ref}}(t)} \tag{17}$$

### 2-3- کنترل مدل پیش‌بین به عنوان کنترل پسخور

روابط (17,16) کنترل‌های حلقه باز سیستم را به‌دست می‌آورند که در شرایط ایده‌آل اطمینان می‌دهد ربات بر مسیر مرجع خود حرکت کند. منظور از شرایط ایده‌آل آن است که مدل سینماتیکی به درستی حرکت ربات را توصیف کند و در آن اغتشاش، خطای اندازه‌گیری یا خطای شرایط اولیه وجود نداشته باشد. این فرضیات هیچ موقع به طور کامل پیش نمی‌آید؛ بنابراین استفاده از کنترل پسخور ضروری است.

در این مقاله به منظور پیش‌بینی رفتار آینده سیستم به محاسبه کنترل پسخور از روش کنترل مدل پیش‌بین می‌پردازیم.

ابتدا مدل خطای سیستم را توضیح می‌دهیم. به منظور حل مسأله کنترلی، تبدیل مختصات ربات به صورتی که برای اهداف کنترلی مناسب‌تر است انجام می‌شود. وضعیت خطا در مختصات جهانی سیستم داده نشده است، اما خطا در مختصات محلی ربات که هم‌راستا با مکانیزم داخلی حرکت ربات را می‌توانیم محاسبه کنیم. این خطا به صورت انحراف ربات مرجع مجازی[1] و ربات واقعی تعریف می‌شود. خطاهای به‌دست‌آمده به شرح زیر است: $e_x$ خطا در مسیر حرکت، $e_y$ خطا عمود بر مسیر و $e_\varphi$ خطای جهت‌گیری که در شکل 2 نشان داده شده است [22]. با استفاده از وضعیت ربات واقعی $q(t) = [x(t), y(t), \varphi(t)]^{\mathrm{T}}$ و وضعیت مرجع ربات مجازی به صورت $q_{\mathrm{ref}}(t) = [x_{\mathrm{ref}}(t), y_{\mathrm{ref}}(t), \varphi_{\mathrm{ref}}(t)]^{\mathrm{T}}$ وضعیت خطا را به صورت رابطه (18) به‌دست می‌آوریم.

$$\begin{bmatrix} e_x\,(t) \\ e_y\,(t) \\ e_\varphi\,(t) \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(q_{\mathrm{ref}}(t) - q(t)\right) \tag{18}$$

فرض می‌کنیم ربات واقعی و ربات مرجع مدل سینماتیکی یکسانی دارند که در رابطه (10) داده شده است. توسط رابطه (18) مدل خطا به صورت رابطه (19) نوشته می‌شود.

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\varphi \end{bmatrix} = \begin{bmatrix} \cos e_\varphi & 0 \\ \sin e_\varphi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{\mathrm{ref}} \\ \omega_{\mathrm{ref}} \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} u \tag{19}$$

$v_{\mathrm{ref}}$ و $\omega_{\mathrm{ref}}$ سرعت‌های مرجع خطی و زاویه‌ای است که در روابط (17,16) داده شده‌اند.

ورودی $u$ توسط کنترل‌کننده به ربات اعمال می‌شود. اغلب $u$ به صورت رابطه (20) تجزیه می‌شود [8].

$$u = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_{\mathrm{ref}} \cos e_\varphi + v_{\mathrm{fb}} \\ \omega_{\mathrm{ref}} + \omega_{\mathrm{fb}} \end{bmatrix} \tag{20}$$

در آن $v_{\mathrm{fb}}$ و $\omega_{\mathrm{fb}}$ ورودی‌های کنترلی پسخور است در حالی که $v_{\mathrm{ref}}\cos e_\varphi$ و $\omega_{\mathrm{ref}}$ ورودی‌های کنترلی پیشخور هستند. $v_{\mathrm{ref}}\cos e_\varphi$ با خطای جهت‌گیری تلفیق شده است، به عبارت دیگر هنگامی که خطای جهت‌گیری به سمت صفر رود یک پیشخور واقعی داریم. با جایگزین کردن رابطه (20) در رابطه (19) مدل خطای ردیابی مسیر به صورت رابطه (21) به‌دست می‌آید.

$$\begin{aligned} \dot{e}_x &= \omega_{\mathrm{ref}} e_y - v_{\mathrm{fb}} + e_y \omega_{\mathrm{fb}} \\ \dot{e}_y &= -\omega_{\mathrm{ref}} e_x + v_{\mathrm{ref}} \sin e_\varphi - e_x \omega_{\mathrm{fb}} \\ \dot{e}_\varphi &= -\omega_{\mathrm{fb}} \end{aligned} \tag{21}$$

هدف کنترلی این است که خطاهای رابطه (21) با انتخاب مناسب $v_{\mathrm{fb}}$ و $\omega_{\mathrm{fb}}$ به سمت صفر همگرا شوند.

مدل خطا رابطه (21) غیرخطی است. قصد داریم این مدل را حول نقطه تعادل خطی‌سازی کنیم. به وضوح نقطه تعادل $(e_x = e_y = e_\varphi = 0)$ است. خطی‌سازی رابطه (21) حول نقطه تعادل به صورت معادلات رابطه (22) به‌دست می‌آید.

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\varphi \end{bmatrix} = \begin{bmatrix} 0 & \omega_{\mathrm{ref}} & 0 \\ -\omega_{\mathrm{ref}} & 0 & v_{\mathrm{ref}} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\varphi \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v_{\mathrm{fb}} \\ \omega_{\mathrm{fb}} \end{bmatrix} \tag{22}$$

که یک سیستم خطی متغیر با زمان است، چون $v_{\mathrm{ref}}$ به و $\omega_{\mathrm{ref}}$ به زمان وابسته است.

ترکیب ورودی کنترلی پیشخور و ورودی کنترلی پسخور به عنوان بردار کنترلی $u$ به صورت رابطه (23) به‌دست می‌آید.

$$u = u_{\mathrm{ff}} + u_{\mathrm{fb}} = \begin{bmatrix} v_{\mathrm{ref}} \cos e_\varphi + v_{\mathrm{fb}} \\ \omega_{\mathrm{ref}} + \omega_{\mathrm{fb}} \end{bmatrix} \tag{23}$$

به طوری که بردار ورودی کنترلی پیشخور $u_{\mathrm{ff}} = [v_{\mathrm{ref}} \cos e_\varphi, \omega_{\mathrm{ref}}]^{\mathrm{T}}$ توسط روابط (17,16) از مسیر مرجع محاسبه می‌شوند و بردار کنترلی $u_{\mathrm{fb}} = [v_{\mathrm{fb}}, \omega_{\mathrm{fb}}]^{\mathrm{T}}$ خروجی کنترل‌کننده مدل پیش‌بین است.

مدل دینامیک خطای ردیابی خطی‌سازی شده به فرم فشرده به صورت رابطه (24) است.

$$\dot{e} = A_c(t)e + B_c u_{\mathrm{fb}} \tag{24}$$

$A_c$ و $B_c$ ماتریس‌های مدل فضای حالت پیوسته و $e$ خطای ردیابی است که توسط رابطه (18) به‌دست می‌آید. وضعیت خطا در مختصات محلی در شکل 2 نشان داده شده است.

روش کنترل مدل پیش‌بین زمان گسسته در [20] به صورت خلاصه ارائه شده است. مدل رابطه (24) به فرم زمان گسسته به صورت رابطه (25) بازنویسی می‌شود.

$$e(k+1) = A(k)e(k) + Bu_{\mathrm{fb}}(k) \tag{25}$$

در آن $n$ تعداد متغیرهای حالت و $A(k) \in \mathbb{R}^n \times \mathbb{R}^n$، $m$ تعداد ورودی‌ها و $B(k) \in \mathbb{R}^n \times \mathbb{R}^m$ است. ماتریس‌های گسسته $A(k)$ و $B$ به صورت رابطه

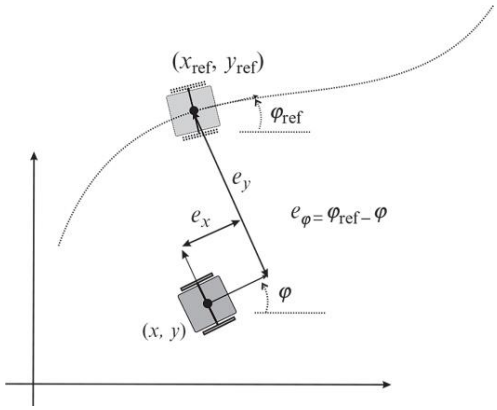


**Fig. 2** Posture error in local coordinates

**شکل 2** وضعیت خطا در مختصات‌های محلی

[1] Virtual reference robot

(26) به‌دست می‌آید.

$$A(k) = \mathrm{I} + A_c(t)T_\mathrm{s}$$
$$B = B_c T_\mathrm{s} \tag{26}$$

تقریب مناسبی برای زمان نمونه برداری کوتاه $T_\mathrm{s}$ است. بهینه‌سازی ورودی کنترلی هدف اصلی کنترل مدل پیش‌بین است، به طوری که تابع هدف داده شده در افق پیش‌بینی $h$ کمینه شود. تابع هدف یک تابع هزینه درجه دو به صورت رابطه (27) است.

$$J(u_\mathrm{fb}, k) = \sum_{i=1}^{h} e(k+i|k)^\mathrm{T} Q e(k+i|k) + u_\mathrm{fb}{}^\mathrm{T}(k+i-1) R u_\mathrm{fb}(k+i-1) \tag{27}$$

به طوری که $e(k+i|k)$، خطای سیستم و $u_\mathrm{fb}(k+i-1)$ ورودی کنترلی، $i$، معرف $i$-امین مرحله پیش‌بینی است $(i = 1, \dots, h)$. همچنین $Q$ و $R$ ماتریس های وزنی هستند. برای پیش‌بینی $e(k+i|k)$ مدل خطای رابطه (24) به شرح رابطه (28) عمل می‌کنیم.

$$\begin{aligned}
e(k+1|k) &= A(k)e(k) + B\mathrm{u}_\mathrm{fb}(k)\\
e(k+2|k) &= A(k+1)e(k+1|k) + Bu_\mathrm{fb}(k+1)\\
e(k+i|k) &= A(k+i-1)e(k+i-1|k) + Bu_\mathrm{fb}(k+i-1)\\
e(k+h|k) &= A(k+h-1)e(k+h-1|k) + Bu_\mathrm{fb}(k+h-1)
\end{aligned} \tag{28}$$

پیش‌بینی‌های $e(k+i|k)$ در رابطه (28) وابسته به خطای فعلی $e(k)$، ورودی‌های کنونی و آینده $u_\mathrm{fb}(k+i-1)$، و ماتریس‌های $A(k+i-1)$ و $B$ است.

مدل پیش‌بینی خروجی در زمان لحظه‌ای $h$ را می‌توان به صورت رابطه (29) بازنویسی کرد.

$$e(k+h|k) = \prod_{j=0}^{h-1} A(k+j)e(k) + \sum_{i=1}^{h}\left(\prod_{j=i}^{h-1} A(k+j)\right) Bu_\mathrm{fb}(k+i-1) + Bu_\mathrm{fb}(k+h-1) \tag{29}$$

طبق روابط (29,28) بردار خطای پیش‌بینی ردیابی مسیر به صورت رابطه (30) تعریف می‌شود.

$$E^*(k) = [e(k+1|k)^\mathrm{T}, e(k+2|k)^\mathrm{T}, \dots, e(k+h|k)^\mathrm{T}]^\mathrm{T} \tag{30}$$

به طوری که $U_\mathrm{fb}(k)$ برای تمام بازه پیش‌بینی $E^*$ به صورت بردار کنترلی رابطه (31) به‌دست می‌آید.

$$U_\mathrm{fb}(k) = [u_\mathrm{fb}(k)^\mathrm{T}, u_\mathrm{fb}(k+1)^\mathrm{T}, \dots, u_\mathrm{fb}(k+h-1)^\mathrm{T}]^\mathrm{T} \tag{31}$$

و رابطه (32) به شرح زیر است.

$$\Lambda(k,i) = \prod_{j=i}^{h-1} A(k+j) \tag{32}$$

بردار خطای پیش‌بینی ردیابی ربات را می‌توان به فرم فشرده رابطه (33) نوشت.

$$E^*(k) = F(k)e(k) + G(k)u_\mathrm{fb}(k) \tag{33}$$

به طوری که رابطه (34) به زیر صورت است.

$$F(k) = [A(k), A(k+1)A(k) \dots \Lambda(k,0)]^\mathrm{T} \tag{34}$$

و همچنین رابطه (35) به شرح زیر است.

$$G(k) = \begin{bmatrix} B & 0 & \cdots & 0 \\ A(k+1)B & B & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda(k,1)B & \Lambda(k,2)B & \cdots & B \end{bmatrix} \tag{35}$$

ابعاد $F$ و $G$ به ترتیب $(n.h, n)$ و $(n.h, m.h)$ است.

قصد داریم برای به‌دست آوردن ورودی‌های کنترلی از حل تحلیلی تابع هدف استفاده کنیم [20].

بردار ورودی‌های کنترلی رابطه (31) توسط بهینه‌سازی رابطه (27) به‌دست می‌آید. تابع هدف رابطه (27) به صورت رابطه (36) نوشته می‌شود.

$$J(U_\mathrm{fb}) = E^{*\mathrm{T}}\bar{Q}E^* + U_\mathrm{fb}{}^\mathrm{T}\bar{R}U_\mathrm{fb} \tag{36}$$

کمینه کردن رابطه (36) به صورت رابطه (37) بیان می‌شود.

$$\frac{\partial J}{\partial U_\mathrm{fb}} = 2\bar{Q}G^\mathrm{T}E^{*\mathrm{T}} + 2\bar{R}U_\mathrm{fb} = 0 \tag{37}$$

بردار بهینه کنترلی به صورت رابطه (38) به دست می‌آید.

$$U_\mathrm{fb} = -(G^\mathrm{T}\bar{Q}G + \bar{R})^{-1}G^\mathrm{T}\bar{Q}Fe(k) \tag{38}$$

اکنون رابطه (39) را می‌توانیم به صورت قانون کنترل پسخور مدل پیش‌بین بنویسیم.

$$U_\mathrm{fb} = K_\mathrm{mpc} \cdot e(k) \tag{39}$$

و ماتریس‌های وزنی به صورت رابطه (40) محاسبه می‌شوند.

$$\bar{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q \end{bmatrix}, \bar{R} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix} \tag{40}$$

از رابطه (38) بردارهای کنترلی $U_\mathrm{fb}{}^\mathrm{T}(k+i-1)$ در تمام طول پیش‌بینی $(i = 1, \dots, h)$ به دست می‌آید. با اعمال ورودی کنترلی پسخور در لحظه $k$ فقط اولین $m$ ردیف $U_\mathrm{fb}{}^\mathrm{T}(k)$ به سیستم ربات اعمال می‌شود.

### 3-3- کنترل مدل پیش‌بین با ضرایب وزنی نمایی[1]

اگر در لحظه‌ای از سیستم خطایی رخ دهد، این خطا در نمونه‌های آینده اثرگذار است. برای کاهش این اثر می‌توان از ضرایبی استفاده کرد که تأثیر خطا در افق‌های جلوتر را تضعیف کند. در این قسمت قصد داریم از ضرایب نمایی استفاده کنیم که به صورت $\alpha^{-j}$ تعریف می‌شود. اگر $\alpha < 1$ باشد، $1 \le \alpha^{-j}, j = 0,1,2,\dots,h$ می‌شود، از این‌رو تأکید روی داده‌ها در افق‌های جلوتر نسبت به زمان حال بیشتر می‌شود؛ بنابراین اگر خطایی در سیستم داشته باشیم این خطا به مرور تقویت خواهد شد. در حالی که اگر $\alpha > 1$ باشد، $1 \ge \alpha^{-j}, j = 0,1,2,\dots,h$ می‌شود، در این حالت حساسیت روی زمان کنونی نسبت زمان‌های آینده بیشتر است و در افق‌های جلوتر تأثیر داده‌ها کمتر خواهد شد. اگر در لحظه‌ای خطایی در سیستم داشته باشیم به مرور در افق‌های جلوتر تضعیف خواهد شد. به منظور استفاده در کنترل مدل پیش‌بین ضریب $\alpha > 1$ را در نظر می‌گیریم.

به ازای $\alpha > 1$ که روی ضرایب وزنی اعمال می‌شود، مقادیر ویژه حلقه بسته بزرگ می‌شوند و این امر سبب جابه‌جایی قطب‌ها و نزدیک شدن آن‌ها به دایره واحد و یا خارج شدن از آن می‌شود. این امر سبب می‌شود پایداری حلقه بسته سیستم به خطر بیافتد، از این‌رو ماتریس‌های $Q$ و $R$ را برای رفع اثر بزرگ شدن مقادیر ویژه طوری تغییر می‌دهیم که مقادیر ویژه حلقه بسته در $\alpha^{-1}$ ضرب شده و اثر $\alpha$ خنثی شود [23].

با توجه به روابط اثبات شده در [23] $Q_a$ و $R_a$ به صورت روابط (43-41) تعریف می‌شوند.

$$\gamma = \frac{1}{\alpha} \tag{41}$$

$$Q_a = \gamma^2 + \left(1 - \gamma^2\right)P_\infty \tag{42}$$

$$R_a = \gamma^2 R \tag{43}$$

[1] Exponential data weighting

$P_\infty$ پاسخ معادله ریکاتی جبری رابطه (44) است.

$$A^{\mathrm{T}}[P_\infty - P_\infty B(R + B^{\mathrm{T}}P_\infty B)^{-1}B^{\mathrm{T}}P_\infty]A + Q - P_\infty = 0 \tag{44}$$

در این روش تابع هدف به صورت رابطه (45) تعریف می‌شود.

$$J(u_{fb},k) = \sum_{i=1}^{h} \alpha^{-2i} e(k+i|k)^{\mathrm{T}}\, Q_a e(k+i|k) + \alpha^{-2(j-1)} {u_{\mathrm{fb}}}^{\mathrm{T}}(k+i-1) R_a u_{\mathrm{fb}}(k+i-1) \tag{45}$$

که در واقع روابط (47,46) به صورت زیر تعریف می‌شوند.

$$E^*(k) = [\alpha^{-1}e(k+1|k)^{\mathrm{T}}, \alpha^{-2}e(k+2|k)^{\mathrm{T}}, \dots, \alpha^{-h}e(k+h|k)^{\mathrm{T}}]^{\mathrm{T}} \tag{46}$$

$$U_{\mathrm{fb}}(k) = [\alpha^{-0}u_{\mathrm{fb}}(k)^{\mathrm{T}}, \alpha^{-1}u_{\mathrm{fb}}(k+1)^{\mathrm{T}}, \dots, \alpha^{-(h-1)}u_{\mathrm{fb}}(k+h-1)^{\mathrm{T}}]^{\mathrm{T}} \tag{47}$$

## 4-3- کنترل مدل پیش‌بین با استفاده از توابع لاگر[1]

یکی از ویژگی‌های مهم کنترل پیش‌بین مدل فضای حالت این است که می‌توانیم سیگنال‌های کنترلی را به صورت ترکیب خطی از توابع پایه در نظر بگیریم. توابع لاگر یکی از توابع معروف است که در کنترل پیش‌بین مدل فضای حالت به وفور استفاده می‌شود و به صورت رابطه (48) تعریف می‌شود [23].

$$\Gamma_k(\mathrm{z}) = \Gamma_{k-1}(\mathrm{z})\frac{\mathrm{z}^{-1} - a}{1 - a\mathrm{z}^{-1}} \tag{48}$$

به طوری که رابطه (49) را به صورت زیر داریم.

$$\Gamma_1(\mathrm{z}) = \frac{\sqrt{1-a^2}}{1 - a\mathrm{z}^{-1}} \tag{49}$$

ویژگی دیگر تابع لاگر این است که هم در حوزه فرکانس و هم در حوزه زمان می‌توان برای آن رابطه بازگشتی به صورت روابط (51,50) به‌دست آورد.

$$L(k) = [l_1(k)\ \ l_2(k)\ \dots\ l_N(k)]^{\mathrm{T}} \tag{50}$$

$$L(k+1) = A_l L(k) \tag{51}$$

افق کنترل از روش کنترل مدل پیش‌بین برای کاهش محاسبات حذف می‌شود و دو پارامتر $N$ و $a$ جایگزین آن می‌شود. به طوری که $N$ تعداد توابع لاگر و $a$ قطب شبکه است، برای این‌که شبکه پایدار باشد باید $0 \le a < 1$ باشد.

با در نظر گرفتن دو تابع لاگر شرایط اولیه به صورت رابطه (52) به دست می‌آید.

$$L(0) = \sqrt{\beta}\begin{bmatrix}1\\-a\end{bmatrix}, \qquad \beta = 1 - a^2 \tag{52}$$

ماتریس $A_l$ یک ماتریس $N \times N$ به صورت رابطه (53) است.

$$A_l = \begin{bmatrix} a & 0 \\ \beta & a \end{bmatrix} \tag{53}$$

در این الگوریتم متغیرهای بهینه‌سازی ضرایب توابع لاگر است و سیگنال کنترلی بهینه از ترکیب خطی توابع لاگر به صورت رابطه (54) به دست می‌آید.

$$u(k_i + k) = \sum_{j=1}^{N} c_j(k_i) l_j(k) \tag{54}$$

به طوری که $k_i$ زمان حاضر است، $k$ گام‌های روبه جلو را نشان می‌دهد.

رابطه (54) را می‌توان به فرم ماتریسی رابطه (55) نوشت.

$$u(k_i + k) = L(k)^{\mathrm{T}}\eta \tag{55}$$

به طوری که رابطه (56) را به صورت زیر داریم.

$$\eta = [c_1\ c_2\ \dots\ c_N]^{\mathrm{T}} \tag{56}$$

اکنون قصد داریم توابع لاگر را به سیستم‌های چند ورودی- چند خروجی بسط دهیم؛ بنابراین ورودی به صورت رابطه (57) بیان می‌شود.

$$u = [u_1, u_2, \dots, u_m] \tag{57}$$

$m$ تعداد ورودی‌ها و $u_i$، $i$-امین ستون ماتریس $u$ است.

اکنون می‌توان ورودی‌ها را به صورت رابطه (58) نوشت، به طوری که هر ورودی به صورت مستقل یک مکان برای قطب لاگر دارد.

$$u_i(k) = L_i(k)^{\mathrm{T}}\eta_i \tag{58}$$

به طوری که در آن $L_i(k) = [{l^i}_1(k)\ \ {l^i}_2(k)\ \dots\ {l^i}_N(k)]^{\mathrm{T}}$ و $\eta_i = [c_1\ c_2\ \dots\ c_N]^{\mathrm{T}}$ است.

به منظور کوتاه‌تر شدن معادلات روابط (60,59) را تعریف می‌کنیم.

$$L_t = \begin{bmatrix} L_1(k)^{\mathrm{T}} & {o_2}^{\mathrm{T}} & \cdots & {o_k}^{\mathrm{T}} \\ {o_1}^{\mathrm{T}} & L_2(k)^{\mathrm{T}} & \cdots & {o_k}^{\mathrm{T}} \\ \vdots & \vdots & \ddots & \vdots \\ {o_1}^{\mathrm{T}} & {o_2}^{\mathrm{T}} & \cdots & L_k(k)^{\mathrm{T}} \end{bmatrix} \tag{59}$$

$$\eta_t = [{\eta_1}^{\mathrm{T}}\ {\eta_2}^{\mathrm{T}} \dots {\eta_k}^{\mathrm{T}}]^{\mathrm{T}} \tag{60}$$

به طوری که $o^{\mathrm{T}}$ بردار صفر است. اکنون می‌توان دنباله ورودی کنترلی سیستم را به صورت رابطه (61) بنویسیم.

$$u(k_i + k) = {L(k)_t}^{\mathrm{T}}\eta_t \tag{61}$$

از این‌رو تابع هزینه به صورت رابطه (63,62) تعریف می‌شود.

$$J(\eta) = {E^*}^{\mathrm{T}}\bar{Q}E^* + {\eta_t}^{\mathrm{T}}R_L\eta_t \tag{62}$$

$$E^*(k) = F(k)e(k) + \Phi(k)\eta_t(k) \tag{63}$$

چون به منظور کاهش محاسبات دو تابع لاگر در نظر گرفتیم، $\Phi(k)$ به صورت رابطه (64) به دست می‌آید.

$$\Phi(k) = \begin{bmatrix} {BL_t}^{\mathrm{T}}(0) \\ A(k+1){BL_t}^{\mathrm{T}}(0) + {BL_t}^{\mathrm{T}}(1) \\ \vdots \\ \Lambda(k,1){BL_t}^{\mathrm{T}}(0) + \Lambda(k,2){BL_t}^{\mathrm{T}}(1) \end{bmatrix} \tag{64}$$

که بعد $\Phi(k)$، $(n.h \times m.N)$ است، همچنین $F(k)$ از رابطه (34) به دست می‌آید.

با کمینه کردن تابع هزینه رابطه (62)، $\eta_t$ به صورت رابطه (65) به دست می‌آید.

$$\eta_t = -(\Phi^{\mathrm{T}}\bar{Q}\Phi + R)^{-1}\Phi^{\mathrm{T}}\bar{Q}Fe(k) \tag{65}$$

براساس قانون کنترل افق بازگشتی اولین $m$ ردیف ورودی کنترلی به صورت رابطه (66) به سیستم ربات اعمال می‌شود.

$$u(k_i) = \begin{bmatrix} {L(0)_1}^{\mathrm{T}} & o^{\mathrm{T}} \\ o^{\mathrm{T}} & {L(0)_2}^{\mathrm{T}} \end{bmatrix}\eta_t \tag{66}$$

## 4- نتایج شبیه‌سازی

قصد داریم در این مقاله مسیر مرجع رابطه (67) را ردیابی کنیم.

$$x_{\mathrm{r}} = 1.1 + 0.7\sin\left(\frac{2\pi t}{30}\right), y_{\mathrm{r}} = 0.9 + 0.7\sin\left(\frac{4\pi t}{30}\right) \tag{67}$$

که $Q = \begin{bmatrix} 9 & 0 & 0 \\ 0 & 90 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$ و $R = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$ تعریف می‌شوند. همچنین نقطه شروع اولیه به صورت $x(0) = [1.09, 0.8, 0]^{\mathrm{T}}$، زمان نمونه برداری $T_{\mathrm{s}} = 0.033$، افق پیش‌بینی $h = 4$، ضریب وزنی نمایی $\alpha = 1.2$ و قطب‌های توابع لاگر هر دو $a_{1,2} = 0.52$ در نظر گرفته شده است.

شبیه‌سازی ردیابی مسیر مرجع توسط کنترل‌کننده مدل پیش‌بین معمولی در شکل 3 و خطای حالت‌های به‌دست‌آمده با استفاده از این کنترل‌کننده در شکل 4، ردیابی مسیر مرجع توسط کنترل‌کننده مدل

[1] Laguerre functions

پیش‌بین با ضرایب وزنی نمایی در شکل 5 و خطاهای متغیرهای حالت منتسب به این کنترل‌کننده در شکل 6 و همچنین ردیابی مسیر مرجع توسط کنترل‌کننده مدل پیش‌بین با توابع لاگر در شکل 7 و خطاهای متغیرهای حالت این کنترل‌کننده در شکل 8 ارائه شده است. برای مقایسه بهتر، خطای حالت‌های به‌دست‌آمده در شکل‌های 4، 6 و 8 به طور جداگانه در دو ثانیه اول در شکل‌های 9، 10 و 11 مقایسه شده‌اند. همچنین در جدول 1 میانگین مربعات خطای حالت‌های سیستم با هم مقایسه شده‌اند.

شرایط اولیه به صورت $x(0) = [1.09\,, 0.8\,, 0]^{\mathrm{T}}$ در نظر گرفته شده است که دو مقدار اول، موقعیت $x$ و موقعیت $y$ در لحظه ابتدایی و مقدار سوم، جهت‌گیری اولیه $\varphi$ را نشان می‌دهد. به همین دلیل ربات ابتدا با زاویه صفر درجه نسبت به محور افقی حرکت می‌کند.

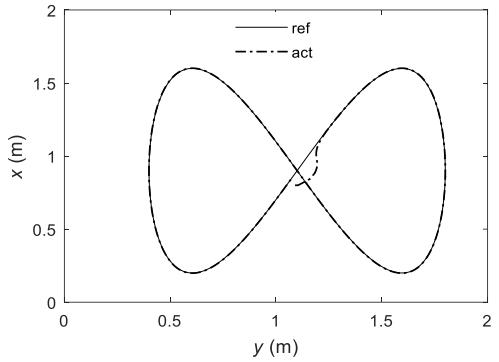


**Fig. 3** Reference trajectory tracking using model predictive control

**شکل 3** ردیابی مسیر مرجع با استفاده از کنترل مدل پیش‌بین

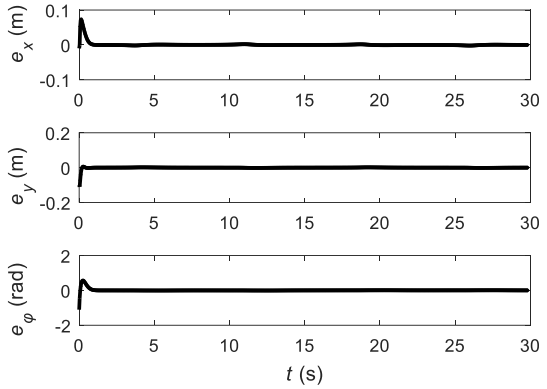


**Fig. 4** States error using model predictive control

**شکل 4** خطای حالت‌های به‌دست‌آمده با استفاده از کنترل مدل پیش‌بین

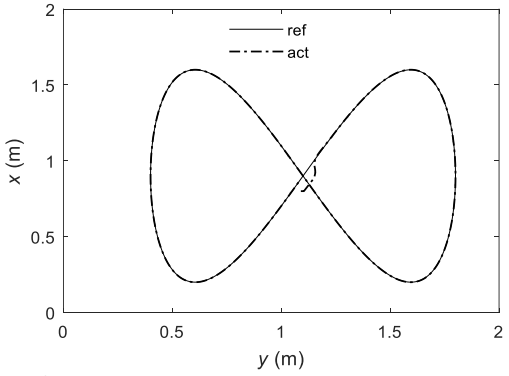


**Fig.5** Reference trajectory tracking using model predictive control with exponential data weighting

**شکل 5** ردیابی مسیر مرجع با استفاده از کنترل پیش‌بین با ضرایب وزنی نمایی

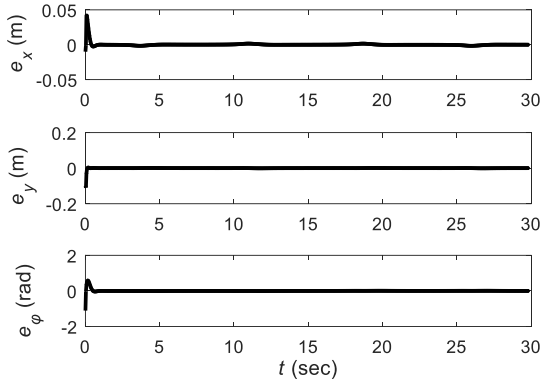


**Fig. 6** States error using model predictive control with exponential data weighting

**شکل 6** خطای حالت‌های به‌دست‌آمده با استفاده از کنترل مدل پیش‌بین با ضرایب وزنی نمایی

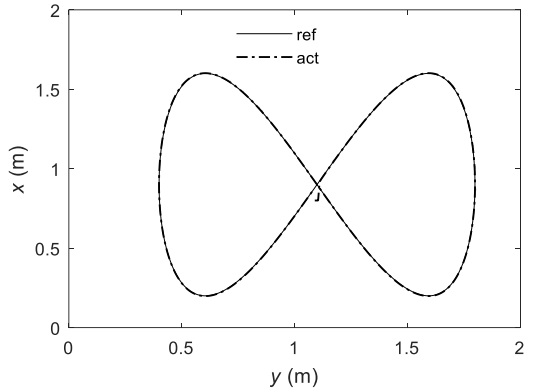


**Fig. 7** Reference trajectory tracking using model predictive control with laguerre functions

**شکل 7** ردیابی مسیر مرجع با استفاده از کنترل پیش‌بین با توابع لاگر

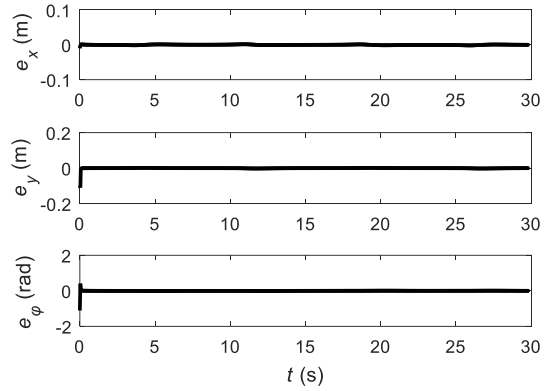


**Fig. 8** States error using model predictive control with laguerre functions

**شکل 8** خطای حالت‌های به‌دست‌آمده با استفاده از کنترل مدل پیش‌بین با توابع لاگر

## 5- بحث و نتیجه‌گیری

برای کنترل ردیابی مسیر موفق ربات متحرک چرخ‌دار دیفرانسیلی نیاز به استفاده از کنترل‌کننده ناهموار یا متغیر با زمان است، زیرا سیستم غیرخطی و متغیر با زمان است. در این مقاله از ساختار کنترلی پیشخور و پسخور استفاده شده است. کنترل پیشخور از مسیر مرجع محاسبه می‌شود و به تنهایی کارا نیست، زیرا قابلیت جبران خطاهای اندازه‌گیری و خطاهای حالت

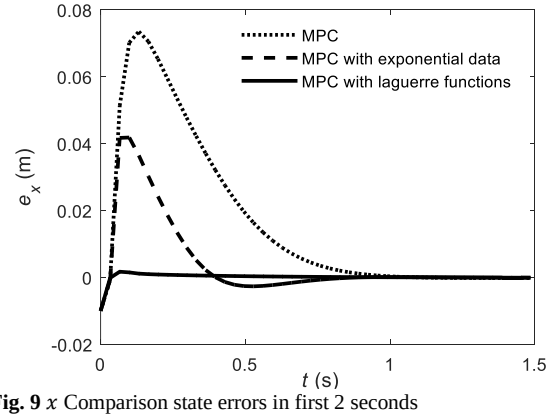


**Fig. 9** $x$ Comparison state errors in first 2 seconds

**شکل 9** مقایسه خطاهای حالت $x$ در 1.5 ثانیه اول

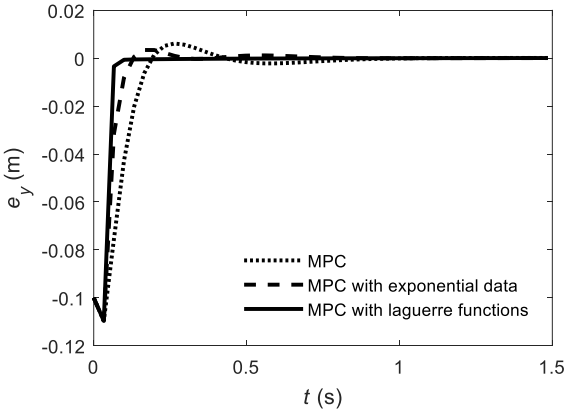


**Fig. 10** $y$ Comparison state errors in first 2 seconds

**شکل 10** مقایسه خطاهای حالت $y$ در 1.5 ثانیه اول

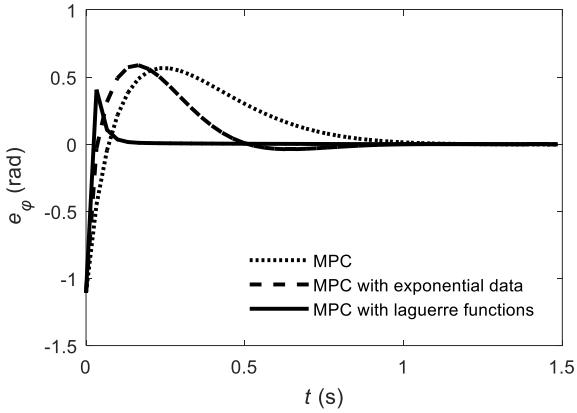


**Fig. 11** $\varphi$ Comparison state errors in first 2 seconds

**شکل 11** مقایسه خطاهای حالت $\varphi$ در 1.5 ثانیه اول

**جدول 1** میانگین مربعات خطای حالت‌ها در طول ردیابی مسیر

**Table 1** RMS states error on trajectory tracking

| کنترل‌کننده | خطا | $i=20$ | $i=50$ | تمام $i$ ها |
|---|---|---|---|---|
| | $e_x$ | 0.0434 | 0.0275 | 0.0065 |
| پیش‌بین معمولی | $e_y$ | 0.0389 | 0.0246 | 0.0058 |
| | $e_\varphi$ | 0.4674 | 0.2976 | 0.0700 |
| | $e_x$ | 0.0188 | 0.0119 | 0.0029 |
| پیش‌بین با ضرایب وزنی نمایی | $e_y$ | 0.0340 | 0.0215 | 0.0051 |
| | $e_\varphi$ | 0.4040 | 0.2557 | 0.0602 |
| | $e_x$ | 0.0024 | 0.0015 | 0.0002 |
| پیش‌بین با توابع لاگر | $e_y$ | 0.0332 | 0.0210 | 0.0050 |
| | $e_\varphi$ | 0.2650 | 0.1676 | 0.0396 |

اولیه را ندارد. در این مقاله برای محاسبه کنترل پسخور از روش کنترل مدل پیش‌بین استفاده شده است. سپس برای بهبود نتایج ردیابی و کم کردن خطای حالت‌های سیستم از کنترل مدل پیش‌بین ضرایب وزنی نمایی و همچنین کنترل مدل پیش‌بین با توابع لاگر استفاده شده است.

با توجه به محاسبات و شبیه‌سازی‌های انجام شده در شکل‌های 4 و 6 و 8 مشاهده می‌کنیم خطای حالت‌های سیستم در تمام مراحل به صفر همگرا می‌شود. برای مقایسه بهتر در شکل‌های 9 و 10 و 11 مشاهده می‌کنیم وقتی از کنترل پیش‌بین با توابع لاگر استفاده می‌کنیم خطای سه حالت زودتر نسبت به کنترل پیش‌بین با ضرایب وزنی نمایی و کنترل پیش‌بین معمولی به صفر همگرا شده است، همچنین در جدول 1 میانگین مربعات خطا در سه مرحله برای بیستمین و پنجاهمین تکرار اول و تمام تکرارها محاسبه شده است. همان طور که مشاهده می‌شود در شرایطی که از کنترل مدل پیش‌بین با توابع لاگر استفاده شده است، نسبت به کنترل مدل پیش‌بین با ضرایب وزنی نمایی و کنترل مدل پیش‌بین معمولی میانگین مربعات خطای حالت‌های سیستم در تمام مراحل کم‌تر شده و زودتر به صفر همگرا شده است. از این‌رو ردیابی مسیر مرجع زودتر صورت گرفته است. با توجه به نتایج شبیه‌سازی استفاده از کنترل مدل پیش‌بین با توابع لاگر در مقایسه با کنترل مدل پیش‌بین با ضرایب وزنی نمایی و کنترل مدل پیش‌بین معمولی علاوه‌بر کاهش محاسبات منجر به بهبود ردیابی مسیر مرجع می‌گردد.

## 6- مراجع


[1] A. Keymasi Khalaji, S. A. A. Moosavian, Design and implementation of a fuzzy sliding mode control law for a wheeled robot towing a trailer, *Modares Mechanical Engineering*, Vol. 14, No. 4, pp. 81-88, 2014. (in Persian فارسی)

[2] S.A.A. Moosavian, E. Papadopoulos, on the kinematics of multiple manipulator space free-flyers and their computation, *Field Robotics*, Vol. 15, No. 4, pp. 207-216, 1998.

[3] D. Atta, B. Subudhi, Decentralized formation control of multiple autonomous underwater vehicles, *Robotics and Automation*, Vol. 28, No. 4, pp. 303-310, 2013.

[4] Y. Dai , S. G. Lee, Formation control of mobile robots with obstacle avoidance based on GOACM using onboard sensors, *Control, Automation and Systems*, Vol. 12, No. 5, pp. 1077-1089, 2014.

[5] I. Kolmanovsky, N. H. McClamroch, Developments in nonholonomic control problems, *IEEE Control Systems*, Vol. 15, No. 6, pp. 20-36, 1995.

[6] R. W. Brockett, Asymptotic stability and feedback stabilization, *Differential Geometric Control Theory*, Vol. 27, No. 1, pp. 181-191. 1983.

[7] G. Oriolo, A. De Luca, M. Vendittelli, WMR control via dynamic feedback linearization design, implementation, and experimental validation, *IEEE Transactions on Control Systems Technology*, Vol. 10, No. 6, pp. 835-852, 2002.

[8] Y. Kanayama, Y. Kimura, F. Miyazaki, T. Noguchi, A stable tracking control method for an autonomous mobile robot, *Robotics and Automation, 1990 Proceedings, 1990 IEEE International Conference on*, pp. 384-389, 1990.

[9] C. Samson, Time-varying feedback stabilization of car-like wheeled mobile robots, *Robotics Research*, Vol. 12, No. 1, pp. 55-64, 1993.

[10] F. Pourboghrat, M. P. Karlsson, Adaptive control of dynamic mobile robots with nonholonomic constraints, *Computers & Electrical Engineering*, Vol. 28, No. 4, pp. 241-253, 2002.

[11] F. M. Raimondi, M. Melluso, A new fuzzy robust dynamic controller for autonomous vehicles with nonholonomic constraints, *Robotics and Autonomous Systems*, Vol. 52, No. 2, pp. 115-131, 2005.

[12] D. H. Kim, J. H. Oh, Tracking control of a two-wheeled mobile robot using input–output linearization, *Control Engineering Practice*, Vol. 7, No. 3, pp. 369-373, 1999.

[13] S. P. M. Noijen, P. F. Lambrechts, H. Nijmeijer, An observer-controller combination for a unicycle mobile robot, *Control*, Vol. 78, No. 2, pp. 81-87, 2005.

[14] S. Blažič, A novel trajectory-tracking control law for wheeled mobile robots, *Robotics and Autonomous Systems*, Vol. 59, No. 11, pp. 1001-1007, 2011.

[15] N. Sarkar, X. Yun, V. Kumar, Control of mechanical systems with rolling constraints, Application to dynamic control of mobile robots, *Robotics Research*, Vol. 13, No. 1, pp. 55-69, 1994.

[16] A. AOllero, O. Amidi, Predictive path tracking of mobile robots. Application to the CMU Navlab, *Proceedings of 5th International Conference on Advanced Robotics, Robots in Unstructured Environments, ICAR*, Vol. 91, pp. 1081-1086, 1991.

[17] J. E. N. Rico, J. G. Ortega, E. F. Camacho, A Smith-predictor-based generalised predictive controller for mobile robot path-tracking, *Control*

*Engineering Practice*, Vol. *7*, No. 6, pp. 729-740, 1999.
[18] F. Kuhne, W. F. Lages, J. G. da Silva Jr, Model predictive control of a mobile robot using linearization, *Proceedings of Mechatronics and Robotics*, pp. 525-530, 2004.
[19] D. Gu, H. Hu, Neural predictive control for a car-like mobile robot, *Robotics and Autonomous Systems*, Vol. 39, No. 2, pp. 73-86, 2002.
[20] G. Klančar, I. Škrjanc, Tracking-error model-based predictive control for mobile robots in real time, *Robotics and Autonomous Systems*, Vol. 55, No. 6 , pp. 460-469, 2007.
[21] R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, pp. 30-31, 47-89, Massachusetts: MIT press, 2011.
[22] Y. Kanayama, A. Nilipour, C. A. Lelm, A locomotion control method for autonomous vehicles, *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1315–1317, 1998.
[23] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB®*, pp. 85-108, NewYork: Springer Science & Business Media, 2009.