



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

مجید شهباززاده¹، سید جلیل ساداتی رستمی^{2*}، سارا میناگر³

1- دانشجوی کارشناسی ارشد، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

2- استادیار، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

3- استادیار، مهندسی برق، دانشگاه صنعتی نوشیروانی بابل

* صندوق پستی 47148-71167، j.sadati@nit.ac.ir

چکیده

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

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 Shahbazzadeh¹, Seyed Jalil Sadati Rostami^{2*}, Sara Minagar³

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

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 علاوه بر 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

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 متحرک¹ 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

¹ Mobile robots

Please cite this article using:

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)

برای ارجاع به این مقاله از عبارت ذیل استفاده نمایید:

2- مدل سینماتیکی و محدودیت های آن

محدودیت های سینماتیکی از ساختار ربات و مدل سینماتیکی آن نشأت می گیرند. این محدودیت ها می توانند هولونومیک⁴ یا غیرهولونومیک باشند [5].

1-2- قیدهای هولونومیک

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

$$f(q) = f(q_1, \dots, q_n) = 0 \quad (1)$$

به طوری که f و مشتقاتش تابع هایی پیوسته هستند. قید رابطه (1) می تواند 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 می دهد.

2-2- قیدهای غیرهولونومیک

قیدهای غیرهولونومیک سرعت های ممکن سیستم یا به عبارتی مسیرهای حرکتی ممکن را محدود می کنند که می توان به صورت رابطه (2) نوشت [5].

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

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

قید سینماتیکی رابطه (2) اگر انتگرال پذیر باشد، هولونومیک است، به این معنی که سرعت ها $\dot{q}_1, \dots, \dot{q}_n$ می توانند از معادله رابطه (2) حذف شوند و قید به فرم رابطه (1) به دست آید.

اگر قید رابطه (2) انتگرال پذیر نباشد غیرهولونومیک است. قیدهای غیرهولونومیک فضای دسترسی به سرعت های سیستم را کاهش می دهند. اگر m قید غیرهولونومیک مستقل خطی به صورت رابطه (2) داشته باشیم، بعد فضای سرعت آن $(n - m)$ می شود.

3-2- مدل سینماتیکی

مدل سینماتیکی رابطه بین ورودی های سیستم و رفتار آن بر فضای حالت سیستم را تشریح می کند. در ربات هایی که با سرعت کمی حرکت می کنند، دینامیک ربات به ندرت مورد توجه قرار می گیرد و مسأله مسیریابی به شکل ساده تری در نظر گرفته می شود. استفاده از مدل سینماتیکی برای ربات هایی که با سرعت بالایی حرکت می کنند پاسخ مناسبی ارائه نمی دهد [21].

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

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

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 می آید.

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

ربات متحرک چرخ دار دیفرانسیلی مکانیزم حرکتی بسیار ساده ای دارد.

شده است [5]. براساس قضیه براکت¹ هیچ فیدبک نامتغیر با زمان همواری از متغیرهای حالت نمی توان یافت که سیستم غیرهولونومیک² را حول نقطه تعادل به طور مجانبی پایدار کند [6].

یکی از روش های کنترل ربات متحرک چرخ دار طراحی کنترل کننده غیرخطی است. نخستین بار در [7-9] قانون کنترل غیرخطی براساس تئوری لیاپانوف طراحی شده است. پس از آن محققین بسیاری بر ساختار کنترل کننده غیرخطی ربات متحرک چرخ دار تحقیق کرده اند، مانند به کارگیری کنترل تطبیقی [10]، تعمیم آن به فازی [11]، خطی سازی ورودی- خروجی [12]، کنترل کننده ترکیبی و طراحی روبت گر در [13] و بسیاری دیگر در [14] قانون کنترل غیرخطی براساس تئوری لیاپانوف مشکل ناپیوستگی در خطای جهت گیری را رفع می کند.

می توان برای کنترل بسیاری از ربات های تجاری فقط مدل سینماتیکی را در نظر گرفت، زیرا کنترل کننده داخلی از پیش برای آن طراحی شده است که دینامیک ربات را مدیریت می کند. در غیر این صورت باید یک جبران ساز دینامیکی پیش از اعمال کنترل کننده سینماتیکی برای سیستم پیاده سازی کرد [15].

کنترل مدل پیش بین یکی از روش هایی است که می تواند به ربات متحرک چرخ دار اعمال شود تا مسیر مرجع از پیش تعیین شده ای را دنبال کند. روش کنترل مدل پیش بین معمولاً با مدل سینماتیکی و یا دینامیکی خطی شده سروکار دارد. تعدادی از روش های کنترل مدل پیش بین ربات متحرک چرخ دار در سال های اخیر گزارش شده است. مانند کنترل مدل پیش بین تعمیم یافته در [16]، کنترل مدل پیش بین با پیش بینی کننده اسمیت³ برای مدیریت کردن تأخیر زمانی سیستم [17]، کنترل مدل پیش بین براساس مدل خطی متغیر با زمان در [18]، کنترل کننده مدل پیش بین غیرخطی با شبکه عصبی چندلایه در [19] توصیف شده است. در بسیاری از روش ها قانون کنترل از بهینه سازی تابع هزینه به دست می آید. یک روش برای به دست آوردن قانون کنترل حل تحلیلی آن است. از این رو محاسبات کمتر و سرعت پیاده سازی آن بیشتر می شود [20].

در این تحقیق برای کنترل ربات متحرک چرخ دار از ساختار کنترلی پیشخور و پسخور استفاده شده است. برای محاسبه کنترل پسخور از کنترل کننده مدل پیش بین با حل تحلیلی آن استفاده شده است، همچنین تکنیک وزن دار کردن نمایشی داده ها جهت بهبود ردیابی مسیر مرجع به کار رفته است.

ساختار ادامه این تحقیق به شرح زیر است: در بخش 2 در مورد محدودیت های سینماتیکی به طور مختصر توضیح داده شده و سپس به بحث در مورد مدل سینماتیکی ربات متحرک چرخ دار 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 کنترل کننده مدل پیش بین استفاده شده است، سپس به توصیف کنترل مدل پیش بین با ضرایب وزنی نمایشی و کنترل مدل پیش بین با توابع لاگر به منظور بهبود ردیابی مسیر مرجع پرداخته شده است. در بخش 4 کنترل کننده های به دست آمده بر ربات متحرک چرخ دار شبیه سازی شده و نتایج آن با کنترل کننده مدل پیش بین معمولی مقایسه شده است. در بخش 5 نتیجه گیری مقاله ارائه شده است.

¹ Brockett

² Nonholonomic

³ Smith-predictor

⁴ Holonomic

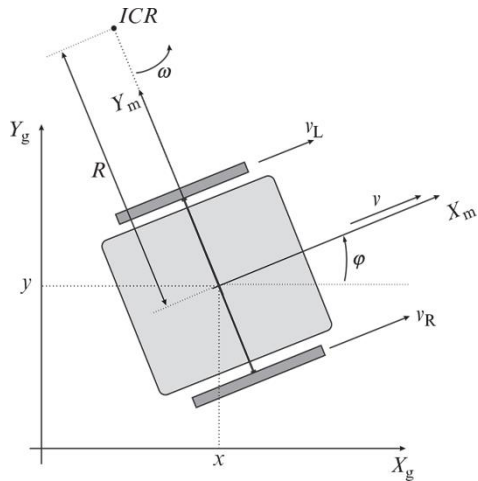


Fig. 1 Differential drive kinematics

شکل 1 سینماتیک رانش دیفرانسیلی

یکی از روش‌های دور زدن محدودیت‌های قضیه برکت این است که از ساختار کنترلی پیشخور^۲ و پسخور^۳ را استفاده کنیم.

3-1- کنترل پیشخور

سیستم ربات متحرک چرخ‌دار مسطح^۴ 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}_u و \mathcal{F}_x به صورت رابطه (12) تعریف می‌شوند.

$$X = \mathcal{F}_x \left(\xi, \dot{\xi}, \ddot{\xi}, \dots, \frac{d^p}{dt^p} \xi \right)$$

$$u = \mathcal{F}_u \left(\xi, \dot{\xi}, \ddot{\xi}, \dots, \frac{d^p}{dt^p} \xi \right) \quad (12)$$

در آن بردارهای X ، u و ξ 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) به دست می‌آید.

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

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

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

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

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

سرعت هر چرخ توسط یک موتور مجزا کنترل می‌شود. نقطه مشترک تقاطع محورهای چرخ‌ها مرکز چرخش آنی^۱ 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.

$$\omega = \frac{v_L(t)}{R(t) - \frac{L}{2}} = \frac{v_R(t)}{R(t) + \frac{L}{2}} \quad (5)$$

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

$$\omega(t) = \frac{v_R(t) - v_L(t)}{L} \quad (6)$$

$$R(t) = \frac{L v_R(t) + v_L(t)}{2 v_R(t) - v_L(t)} \quad (7)$$

سرعت خطی به صورت رابطه (8) محاسبه می‌شود

$$v(t) = \omega(t)R(t) = \frac{v_R(t) + v_L(t)}{2} \quad (8)$$

سرعت‌های خطی چرخ‌های چپ و راست به ترتیب $v_L(t) = r\omega_L(t)$ و $v_R(t) = r\omega_R(t)$ حول محورشان $\omega_L(t)$ و $\omega_R(t)$ است. سینماتیک داخلی ربات (در مختصات محلی) را می‌توان به صورت رابطه (9) بیان کرد.

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} v X_m(t) \\ v Y_m(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ 0 & 0 \\ -\frac{r}{L} & \frac{r}{L} \end{bmatrix} \begin{bmatrix} \omega_L(t) \\ \omega_R(t) \end{bmatrix} \quad (9)$$

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

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

به طوری که v و ω ورودی‌های کنترل هستند. مدل رابطه (10) را می‌توان با تقریب اولر به فرم گسسته زیر نوشت.

زمان نمونه برداری سیستم به صورت رابطه (11) است.

$$x(k+1) = x(k) + v(k)T_s \cos(\varphi(k))$$

$$y(k+1) = y(k) + v(k)T_s \sin(\varphi(k))$$

$$\varphi(k+1) = \varphi(k) + \omega(k)T_s \quad (11)$$

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 واقعی 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)

² Feedforward

³ Feedback

⁴ Flat

(19) مدل خطای ردیابی مسیر به صورت رابطه (21) به دست می آید.

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

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

مدل خطا رابطه (21) غیر خطی است. قصد داریم این مدل را حول نقطه تعادل خطی سازی کنیم. به وضوح نقطه تعادل $(e_x = e_y = e_\varphi = 0)$ است. خطی سازی رابطه (21) حول نقطه تعادل به صورت معادلات رابطه (22) به دست می آید.

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

که یک سیستم خطی متغیر با زمان است، چون v_{ref} و ω_{ref} به زمان وابسته است.

ترکیب ورودی کنترلی پیشخور و ورودی کنترلی پسخور به عنوان بردار کنترلی u به صورت رابطه (23) به دست می آید.

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

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

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

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

A_c و B_c ماتریس های مدل فضای حالت پیوسته و e خطای ردیابی است که توسط رابطه (18) به دست می آید. وضعیت خطا در مختصات محلی در شکل 2 نشان داده شده است.

روش کنترل مدل پیش بین زمان گسسته در [20] به صورت خلاصه ارائه شده است. مدل رابطه (24) به فرم زمان گسسته به صورت رابطه (25) بازنویسی می شود.

$$e(k+1) = A(k)e(k) + B u_{fb}(k) \quad (25)$$

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

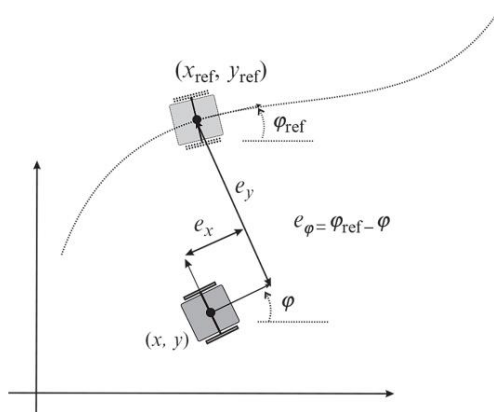


Fig. 2 Posture error in local coordinates

شکل 2 وضعیت خطا در مختصات محلی

در ربات های متحرک چرخ دار دیفرانسیلی سرعت های مرجع $v_{ref}(t)$ و $\omega_{ref}(t)$ از مسیری مرجع توسط $x_{ref}(t)$ و $y_{ref}(t)$ براساس روابط (15,13) به صورت روابط (17,16) به دست آورده می شوند.

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

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

2-3- کنترل مدل پیش بین به عنوان کنترل پسخور

روابط (17,16) کنترل های حلقه باز سیستم را به دست می آورند که در شرایط ایده آل اطمینان می دهد ربات بر مسیر مرجع خود حرکت کند. منظور از شرایط ایده آل آن است که مدل سینماتیکی به درستی حرکت ربات را توصیف کند و در آن اغتشاش، خطای اندازه گیری یا خطای شرایط اولیه وجود نداشته باشد. این فرضیات هیچ موقع به طور کامل پیش نمی آید؛ بنابراین استفاده از کنترل پسخور ضروری است.

در این مقاله به منظور پیش بینی رفتار آینده سیستم به محاسبه کنترل پسخور از روش کنترل مدل پیش بین می پردازیم.

ابتدا مدل خطای سیستم را توضیح می دهیم. به منظور حل مسأله کنترلی، تبدیل مختصات ربات به صورتی که برای اهداف کنترلی مناسب تر است انجام می شود. وضعیت خطا در مختصات جهانی سیستم داده نشده است، اما خطا در مختصات محلی ربات که هم راستا با مکانیزم داخلی حرکت ربات را می توانیم محاسبه کنیم. این خطا به صورت انحراف ربات مرجع مجازی¹ و ربات واقعی تعریف می شود. خطاهای به دست آمده به شرح زیر است: e_x خطا در مسیر حرکت، e_y خطا عمود بر مسیر و e_φ خطای جهت گیری که در شکل 2 نشان داده شده است [22]. با استفاده از وضعیت ربات واقعی $q(t) = [x(t), y(t), \varphi(t)]^T$ و وضعیت مرجع ربات مجازی به صورت $q_{ref}(t) = [x_{ref}(t), y_{ref}(t), \varphi_{ref}(t)]^T$ وضعیت خطا را به صورت رابطه (18) به دست می آوریم.

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

فرض می کنیم 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 می شود.

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

v_{ref} و ω_{ref} سرعت های مرجع خطی و زاویه ای است که در روابط (17,16) داده شده اند.

ورودی u توسط کنترل کننده به ربات اعمال می شود. اغلب u به صورت رابطه (20) تجزیه می شود [8].

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

در آن v_{fb} و ω_{fb} ورودی های کنترلی پسخور است در حالی که $v_{ref} \cos e_\varphi$ و ω_{ref} ورودی های کنترلی پیشخور هستند. با $v_{ref} \cos e_\varphi$ خطای جهت گیری تلفیق شده است، به عبارت دیگر هنگامی که خطای جهت گیری به سمت صفر رود یک پیشخور واقعی داریم. با جایگزین کردن رابطه (20) در رابطه

¹ Virtual reference robot

(26) به دست می‌آید.

$$\begin{aligned} A(k) &= I + A_c(t)T_s \\ B &= B_cT_s \end{aligned} \quad (26)$$

تقریب مناسبی برای زمان نمونه برداری کوتاه T_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.

$$\begin{aligned} J(u_{fb}, k) &= \sum_{i=1}^h e(k+i|k)^T Q e(k+i|k) \\ &\quad + u_{fb}^T(k+i-1) R u_{fb}(k+i-1) \end{aligned} \quad (27)$$

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

$$\begin{aligned} e(k+1|k) &= A(k)e(k) + B u_{fb}(k) \\ e(k+2|k) &= A(k+1)e(k+1|k) + B u_{fb}(k+1) \\ e(k+i|k) &= A(k+i-1)e(k+i-1|k) \\ &\quad + B u_{fb}(k+i-1) \\ e(k+h|k) &= A(k+h-1)e(k+h-1|k) \\ &\quad + B u_{fb}(k+h-1) \end{aligned} \quad (28)$$

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

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

$$\begin{aligned} e(k+h|k) &= \prod_{j=0}^{h-1} A(k+j)e(k) + \sum_{i=1}^h \left(\prod_{j=i}^{h-1} A(k+j) \right) \\ &\quad B u_{fb}(k+i-1) + B u_{fb}(k+h-1) \end{aligned} \quad (29)$$

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

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

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

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

و رابطه (32) به شرح زیر است.

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

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

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

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

$$F(k) = [A(k), A(k+1)A(k) \dots \Lambda(k, 0)]^T \quad (34)$$

و همچنین رابطه (35) به شرح زیر است.

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

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

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

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

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

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

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

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

$$U_{fb} = -(G^T \bar{Q} G + \bar{R})^{-1} G^T \bar{Q} F e(k) \quad (38)$$

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

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

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

$$\bar{Q} = \begin{bmatrix} Q & 0 & \dots & 0 \\ 0 & Q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q \end{bmatrix}, \bar{R} = \begin{bmatrix} R & 0 & \dots & 0 \\ 0 & R & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R \end{bmatrix} \quad (40)$$

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

3-3- کنترل مدل پیش‌بین با ضرایب وزنی نمایی¹

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 α^{-j} تعریف می‌شود. اگر $\alpha < 1$ باشد، $1 \leq \alpha^{-j}, j=0, 1, 2, \dots, h$ می‌شود، از این رو 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، $1 \geq \alpha^{-j}, j=0, 1, 2, \dots, h$ می‌شود، در این حالت حساسیت روی زمان کنونی نسبت به 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 بیفتد، از این رو ماتریس‌های Q و R را برای رفع اثر بزرگ شدن مقادیر ویژه طوری تغییر می‌دهیم که مقادیر ویژه حلقه بسته در α^{-1} ضرب شده و اثر α خنثی شود [23].

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

$$\gamma = \frac{1}{\alpha} \quad (41)$$

$$Q_a = \gamma^2 + (1 - \gamma^2) P_o \quad (42)$$

$$R_a = \gamma^2 R \quad (43)$$

¹ Exponential data weighting

$$\eta = [c_1 \ c_2 \ \dots \ c_N]^T \quad (56)$$

اکنون قصد داریم توابع لاگر را به سیستم های چند ورودی- چند خروجی بسط دهیم؛ بنابراین ورودی به صورت رابطه (57) بیان می شود.

$$u = [u_1, u_2, \dots, u_m] \quad (57)$$

m تعداد ورودی ها و u_i i -امین ستون ماتریس u است. اکنون می توان ورودی ها را به صورت رابطه (58) نوشت، به طوری که هر ورودی به صورت مستقل یک مکان برای قطب لاگر دارد.

$$u_i(k) = L_i(k)^T \eta_i \quad (58)$$

به طوری که در آن $L_i(k) = [l_{i1}^T(k) \ l_{i2}^T(k) \ \dots \ l_{iN}^T(k)]^T$ و $L_i(k) = [c_1 \ c_2 \ \dots \ c_N]^T$ است.

به منظور کوتاه تر شدن معادلات روابط (60,59) را تعریف می کنیم.

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

$$\eta_t = [\eta_1^T \ \eta_2^T \ \dots \ \eta_k^T]^T \quad (60)$$

به طوری که o^T بردار صفر است. اکنون می توان دنباله ورودی کنترلی سیستم را به صورت رابطه (61) بنویسیم.

$$u(k_i + k) = L(k)^T \eta_t \quad (61)$$

از این رو تابع هزینه به صورت رابطه (62,63) تعریف می شود.

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

$$E^*(k) = F(k)e(k) + \Phi(k)\eta_t(k) \quad (63)$$

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

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

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

با کمینه کردن تابع هزینه رابطه (62)، η_t به صورت رابطه (65) به دست می آید.

$$\eta_t = -(\Phi^T \bar{Q} \Phi + R)^{-1} \Phi^T \bar{Q} F e(k) \quad (65)$$

بر اساس قانون کنترل افق بازگشتی اولین m ردیف ورودی کنترلی به صورت رابطه (66) به سیستم ربات اعمال می شود.

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

4- نتایج شبیه سازی

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

$$x_r = 1.1 + 0.7 \sin\left(\frac{2\pi t}{30}\right), y_r = 0.9 + 0.7 \sin\left(\frac{4\pi t}{30}\right) \quad (67)$$

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 می شوند. همچنین

نقطه شروع اولیه به صورت $x(0) = [1.09, 0.8, 0]^T$ ، $T_s = 0.033$ ، افق پیش بینی $h = 4$ ، ضریب وزنی نمای $\alpha = 1.2$ و قطب های توابع لاگر هر دو $\alpha_{1,2} = 0.52$ در نظر گرفته شده است.

شبیه سازی ردیابی مسیر مرجع توسط کنترل کننده مدل پیش بین معمولی در شکل 3 و خطای حالت های به دست آمده با استفاده از این کنترل کننده در شکل 4، ردیابی مسیر مرجع توسط کنترل کننده مدل

P_∞ پاسخ معادله ریکاتی جبری رابطه (44) است.

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

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

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

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

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

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

4-3- کنترل مدل پیش بین با استفاده از توابع لاگر¹

یکی از ویژگی های مهم کنترل پیش بین مدل فضای حالت این است که می توانیم سیگنال های کنترلی را به صورت ترکیب خطی از توابع پایه در نظر بگیریم. توابع لاگر یکی از توابع معروف است که در کنترل پیش بین مدل فضای حالت به وفور استفاده می شود و به صورت رابطه (48) تعریف می شود [23].

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

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

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

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

$$L(k) = [l_1(k) \ l_2(k) \ \dots \ l_N(k)]^T \quad (50)$$

$$L(k+1) = A_l L(k) \quad (51)$$

افق کنترل از روش کنترل مدل پیش بین برای کاهش محاسبات حذف می شود و دو پارامتر N و a جایگزین آن می شود. به طوری که تعداد توابع لاگر و a قطب شبکه است، برای این که شبکه پایدار باشد باید $0 \leq a < 1$ باشد.

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

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

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

$$A_l = \begin{bmatrix} a & 0 \\ \beta & a \end{bmatrix} \quad (53)$$

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

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

به طوری که k_i زمان حاضر است، k گام های روبه جلو را نشان می دهد.

رابطه (54) را می توان به فرم ماتریسی رابطه (55) نوشت.

$$u(k_i + k) = L(k)^T \eta \quad (55)$$

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

¹ Laguerre functions

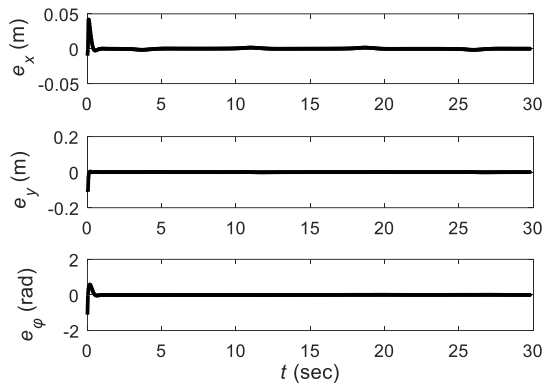


Fig. 6 States error using model predictive control with exponential data weighting

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

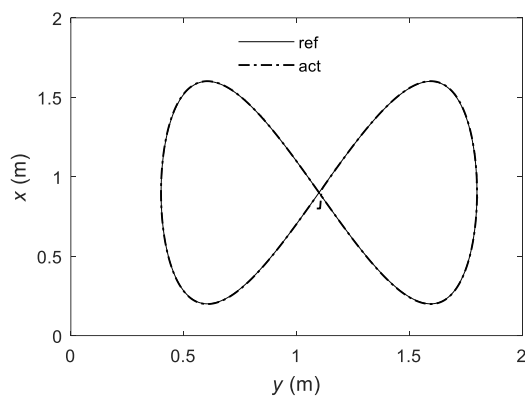


Fig. 7 Reference trajectory tracking using model predictive control with laguerre functions

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

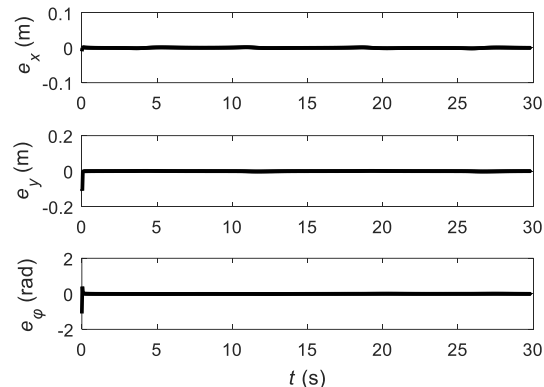


Fig. 8 States error using model predictive control with laguerre functions

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

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

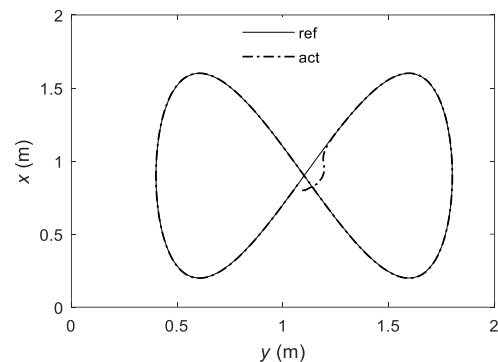


Fig. 3 Reference trajectory tracking using model predictive control

شکل 3 ردیابی مسیر مرجع با استفاده از کنترل مدل پیش‌بین

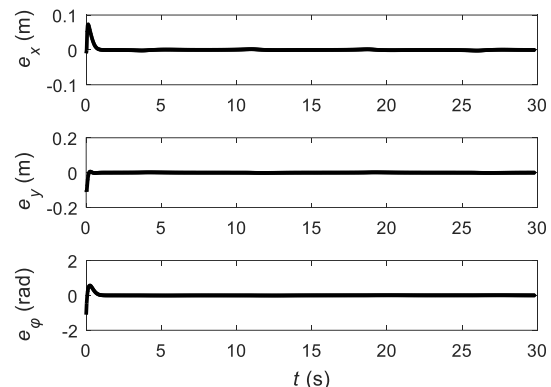


Fig. 4 States error using model predictive control

شکل 4 خطای حالت‌های به‌دست‌آمده با استفاده از کنترل مدل پیش‌بین

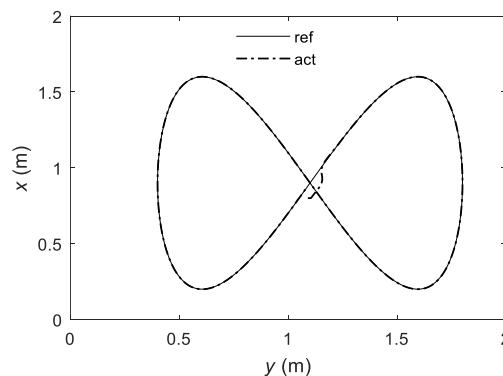


Fig. 5 Reference trajectory tracking using model predictive control with exponential data weighting

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

اولیه را ندارد. در این مقاله برای محاسبه کنترل پسخور از روش کنترل مدل پیش بین استفاده شده است. سپس برای بهبود نتایج ردیابی و کم کردن خطای حالت های سیستم از کنترل مدل پیش بین ضرایب وزنی نمایی و همچنین کنترل مدل پیش بین با توابع لاگر استفاده شده است. با توجه به محاسبات و شبیه سازی های انجام شده در شکل های 4 و 6 و 8 مشاهده می کنیم خطای حالت های سیستم در تمام مراحل به صفر همگرا می شود. برای مقایسه بهتر در شکل های 9 و 10 و 11 مشاهده می کنیم وقتی از کنترل پیش بین با توابع لاگر استفاده می کنیم خطای سه حالت زودتر نسبت به کنترل پیش بین با ضرایب وزنی نمایی و کنترل پیش بین معمولی به صفر همگرا شده است، همچنین در جدول 1 میانگین مربعات خطا در سه مرحله برای بیستیمین و پنجاهمین تکرار اول و تمام تکرارها محاسبه شده است. همان طور که مشاهده می شود در شرایطی که از کنترل مدل پیش بین با توابع لاگر استفاده شده است، نسبت به کنترل مدل پیش بین با ضرایب وزنی نمایی و کنترل مدل پیش بین معمولی میانگین مربعات خطای حالت های سیستم در تمام مراحل کم تر شده و زودتر به صفر همگرا شده است. از این رو ردیابی مسیر مرجع زودتر صورت گرفته است. با توجه به نتایج شبیه سازی استفاده از کنترل مدل پیش بین با توابع لاگر در مقایسه با کنترل مدل پیش بین با ضرایب وزنی نمایی و کنترل مدل پیش بین معمولی علاوه بر کاهش محاسبات منجر به بهبود ردیابی مسیر مرجع می گردد.

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)
- [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. Pourboghra, 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*

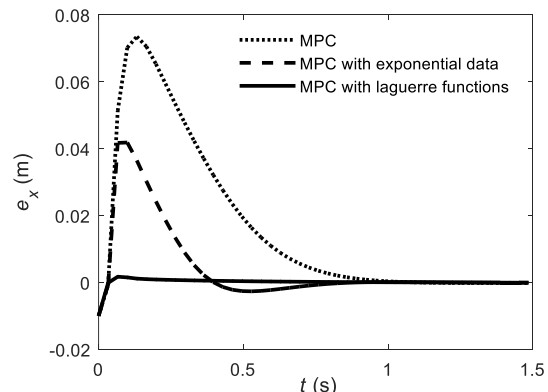


Fig. 9 x Comparison state errors in first 2 seconds

شکل 9 مقایسه خطاهای حالت x در 1.5 ثانیه اول

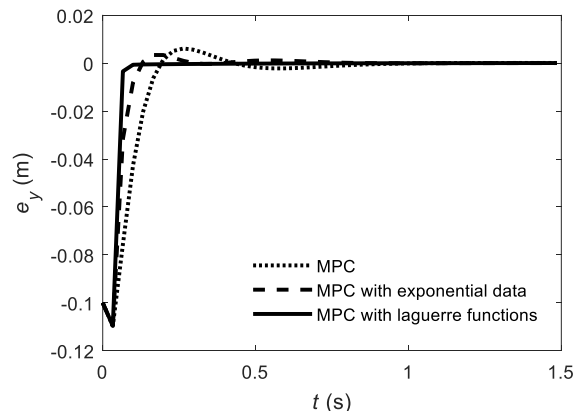


Fig. 10 y Comparison state errors in first 2 seconds

شکل 10 مقایسه خطاهای حالت y در 1.5 ثانیه اول

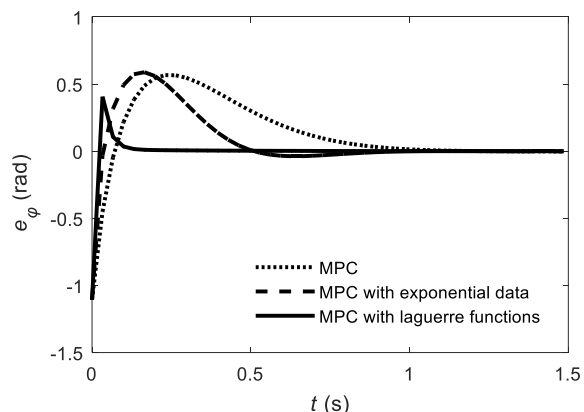


Fig. 11 phi Comparison state errors in first 2 seconds

شکل 11 مقایسه خطاهای حالت phi در 1.5 ثانیه اول

جدول 1 میانگین مربعات خطای حالت ها در طول ردیابی مسیر

Table 1 RMS states error on trajectory tracking			
تمام i ها	i = 50	i = 20	خطا
0.0065	0.0275	0.0434	e_x
0.0058	0.0246	0.0389	e_y
0.0700	0.2976	0.4674	e_ϕ
0.0029	0.0119	0.0188	e_x
0.0051	0.0215	0.0340	e_y
0.0602	0.2557	0.4040	e_ϕ
0.0002	0.0015	0.0024	e_x
0.0050	0.0210	0.0332	e_y
0.0396	0.1676	0.2650	e_ϕ

- 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.
- 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.