طراحی و تعقیب مسیر رباتهای زیرسطحی در محیطهای دینامیک با استفاده از روشهای حل تبدیل هوشمند و کلاسیک

حسن صیادی[۱] ، عبدالرضا باباخانی[۲]

۱- دانشیار دانشکده مهندسی مکانیک، دانشگاه صنعتی شریف
۲- کارشناس ارشد مهندسی مکانیک و مشاور مرکز تحقیقات دریایی بوشهر

چکیده

در این مقاله ابتدا یک نوع ربات زیرسطحی با نام NEROV ساخته شده در دانشگاه Trondheim نروژ انتخاب ومعادلات دینامیکی آن نوشته شده و با استفاده از روش بهینه‌سازی برنامه‌ریزی دینامیک، برای آن یک مسیر بهینه‌ی انرژی طراحی می‌شود. در محیطی که ربات حرکت می‌کند، موانع متحرک وجود دارند. مشاهده می‌شود که مسیر طراحی شده یک مسیر هموار بوده و انرژی مصرفی آن کمینه می‌شود. پس از طراحی مسیر، برای ربات با استفاده از روش اسلاتین و لی، سیستم کنترل خودکار جهت اجرای طی مسیر توسط ربات طراحی می‌شود. در این مقاله، از اثر غیر مستقیم حرکت مانع بر دینامیک ربات صرفنظر شده و فرض شده است که معادلات دینامیک حرکت ربات در محیط تغییری نمی‌کنند. در مرحله آخر نیز، با استفاده از روش حل تبدیل هوشمند، برای ربات طراحی مسیر صورت گرفته و نیز با استفاده از همان روش اسلاتین و لی، عمل تعقیب مسیر صورت پذیرفته است.

کلمات کلیدی: طراحی مسیر، کنترل، موانع متحرک، رباتهای زیرسطحی، برنامه‌ریزی دینامیک، تابع همیلتونی، شبکه‌های عصبی، حل تبدیل هوشمند

PATH PLANNING & TRAJECTORY TRACKING OF AUVS IN DYNAMIC ENVIRONMENTS USING INTELLIGENT CONVERTED SOLUTION AND CLASSICAL METHODS

Hassan Sayyaadi[1], A.Reza Babakhani[2]

1- Associate Professor, Department of Mech. Eng., Sharif University of Technology
2- M.Sc. Graduate, Bushehr Marine Research Center

Abstract

In this paper, first a sample Autonomous Underwater Vehicle called NEROV from the University of Trondheim of Norway has been selected with its equations of motion and by using dynamic programming approach; optimal paths for consuming energy have been designed. There are moving obstacles in the environment wherein the robot moves. It is seen that the planned paths is smooth and the energy used for the robot is minimized. After planning the path, a control system has been designed for by using Slotine&Li algorithm. In this paper, the interaction between the robot dynamics and environmental effects due to the obstacles dynamics is ignored. In the final stage of the path planning, using an Intelligent Converted Solution (ICS), an optimal path is designed for the vehicle and employing the same control system, the optimal path is tracked.

Keywords:Path Planning, Control, Moving Obstacles, Autonomous Underwater Vehicles (AUVs), Dynamic Programming, Hamiltonian Function, Neural Network, Intelligent Converted Solution

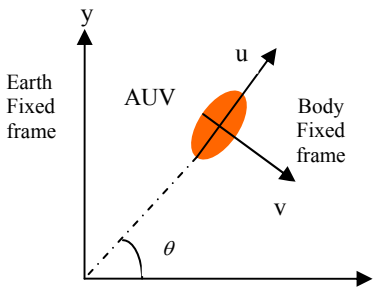
[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

۱- مقدمه

طراحی مسیر رباتها در محیطهای دینامیک، یک کار مشکل می‌باشد. حرکت مانع در محیط باعث پیچیدگی مسئله خواهد شد. امروزه، محیطهایی که در آن موانع حرکت می‌کنند، تئوری‌های مربوطه حتی در زمانهایی که موانع در محیط با سرعت ثابت حرکت می‌کنند نیز، پیچیده می‌باشد. جدای از روشهای عمومی طراحی مسیر همچون خطوط جریان [1]، میدان پتانسیل، تجزیه‌ی سلولی و...[2] از جمله کارهایی که در این زمینه انجام گرفته است، می‌توان به تئوری ریف و شریر [3] اشاره داشت که در محیط، موانع با سرعت ثابت حرکت می‌کنند. همچنین روش فوجیمورا که در این روش سعی شده است ربات مسیرهای کمینه زمانی را در چنین محیطهایی طی کند [4] و یا روشهای دیگر. در این پژوهش، یک نوع از رباتهایی که معادلات حرکت آن پیچیده می‌باشد (رباتهای زیرسطحی) انتخاب و یک مدل نمونه معرفی می‌گردد. در این مقاله هدف این‌است که ربات در محیطی که موانع حرکت می‌کنند، طراحی مسیر شده و در چارچوب مسائل بهینه‌سازی، تابع هدف را کمینه کند. رباتهای زیرسطحی با محدودیت منبع انرژی مواجه‌اند و غالباً از باتریهای قابل شارژ استفاده می‌کنند و یکی از گلوگاههای طراحی و کاربرد اینگونه از ربات‌ها، کاهش میزان مصرف انرژی است. در نتیجه در اینجا تابع هدف، انرژی مصرفی ربات در مدت زمان انجام مأموریت است. روش بهینه‌سازی انتخابی، برنامه‌ریزی دینامیک بوده و روش حل عددی معادلات بدست آمده، نیوتون-رافسون است. در نهایت، مسیر طراحی شده مسیری است که انرژی مصرفی آن کمینه بوده و با مانع متحرک برخوردی ندارد. پس از طراحی مسیر، یک سیستم کنترل خودکار نیز برای تعقیب مسیر ربات انتخاب و طراحی می‌شود. برای این نوع ربات نمونه، سیستم کنترلی انتخابی به روش اسلاتین و لی طراحی شده است. در این مسأله، از اثر غیر مستقیم حرکت مانع بر دینامیک ربات صرفنظر شده و معادلات دینامیک ربات نمونه نیز در طول طراحی و طی مسیر ثابت خواهد بود. با استفاده از حل تبدیل هوشمند نیز، در یک حالت اضافی دیگر که در محیط یک مانع ثابت و یک مانع متحرک حضور دارند، عمل طراحی مسیر صورت پذیرفته و با استفاده از همان سیستم کنترلی نیز عمل تعقیب مسیر صورت پذیرفته است. در حقیقت هر دو مسئله طراحی مسیر (Path Planning) و تعقیب مسیر (Trajectory Tracking) در این مقاله مدنظر قرار گرفته است.

۲- معادلات دینامیکی سیستم

برای رباتهای زیرسطحی می‌توان دو نوع مختصات حرکتی تعریف کرد که یکی مختصات محلی یا مختصات متصل به جسم[1] و دیگری مختصات کلی یا مختصات متصل به زمین[2] می‌باشد. در شکل (۱) این مختصات نشان داده شده است. مختصات نشان داده شده در شکل محدود به صفحه می‌باشد.



شکل ۱- نمای مختصات محلی و کلی در صفحه

معادلات حرکتی رباتهای زیرسطحی در مختصات محلی در حالت ماتریسی و در صفحه بصورت زیر است [5]:

$$M\dot{\vec{v}} + C(\vec{v})\vec{v} + D(\vec{v})\vec{v} = \vec{\tau} \qquad (۱)$$

که در آن: $\vec{\tau} = [X, Y, Z]$ بردار نیروهای خارجی که بترتیب نیروی جلوبرنده، نیرو در جهت عرضی و گشتاور زاویه‌ای می‌باشد و $\vec{v} = [u, v, r]$ بردار سرعت در درجات آزادی مربوطه می‌باشد. u و v در شکل (۱)

[1].Body fixed frame
[2].Earth fixed frame

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

نشان داده شده‌اند و نیز $r = \dot{\theta}$ است. M ماتریس قطری جرم ربات بوده که برای رباتهای زیرسطحی متشکل از جرم و ممان اینرسی خالص و نیز جرم و ممان اینرسی افزوده است. مسلماً برای این نوع از رباتها، مولفه‌های جرم ماتریس M در جهات مختلف متفاوت است، چون جرمهای افزوده متفاوتند [5]. این تفاوت ناشی از فرم بدنه‌ی ربات زیرسطحی در زیر آب است که $C(\vec{v})$ ماتریس تاثیر درجات آزادی بر روی هم بوده و $D(\vec{v})$ نیز ماتریس تأثیر هر درجه آزادی بر روی خود است. می‌توان معادلات دینامیک سیستم را در صفحه از مختصات محلی به مختصات کلی انتقال داد. ماتریس انتقال آن بفرم رابطه (۲) است که به آن ژاکوبین می‌گویند [6]:

$$J(\vec{\eta}) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (۲)$$

در اینصورت معادلات دینامیک سیستم در مختصات متصل به زمین بصورت رابطه (۳) خواهد بود:

$$M_\eta(\vec{\eta})\ddot{\vec{\eta}} + C_\eta(\vec{v},\vec{\eta})\dot{\vec{\eta}} + D_\eta(\vec{v},\vec{\eta})\dot{\vec{\eta}} = J^{-T}(\vec{\eta})\,\vec{\tau} \qquad (۳)$$

که در آن:

(۴)

$$M_\eta(\vec{\eta}) = J^{-T}(\vec{\eta})MJ^{-1}(\vec{\eta})$$
$$C_\eta(\vec{v},\vec{\eta}) = J^{-T}(\vec{\eta})\left[C(\vec{v}) - MJ^{-1}(\vec{\eta})\dot{J}(\vec{\eta})\right]J^{-1}(\vec{\eta})$$
$$D_\eta(\vec{v},\vec{\eta}) = J^{-T}(\vec{\eta})D(\vec{v})J^{-1}(\vec{\eta})$$

ارتباط بین درجات آزادی در دو مختصات تعریف شده مطابق شکل (۱) به‌صورت زیر است:

$$\dot{\vec{\eta}} = J(\vec{\eta})\vec{v}$$

که در آن:

$$\vec{\eta} = [x \;\; y \;\; \theta]$$

لازم به ذکر است که معادلات دینامیک سیستم در مختصات متصل به زمین به‌مراتب پیچیده‌تر از معادلات آن در مختصات متصل به جسم است.

در این تحقیق، یک ربات زیرسطحی با عنوان NEROV از دانشگاه نروژ انتخاب می‌شود [7]. این ربات از نظر آزادی دینامیکی، یک ربات هولونومیک می‌باشد. متحرکهای هولونومیک به متحرکهایی گفته می‌شود که می‌توانند هر مسیری را در مختصات متصل به زمین طی کنند. در این مسئله، فرض شده است که ربات فقط در صفحه حرکت می‌کند.

برای این ربات ماتریسهای M، $C(\vec{v})$ و $D(\vec{v})$ که در رابطه‌ی (۱) مشخص شده‌اند، بترتیب برابرند با:

$$M = \begin{bmatrix} 205 & 0 & 0 \\ 0 & 265 & 0 \\ 0 & 0 & 80 \end{bmatrix} \qquad (۵)$$

$$C(\vec{v}) = \begin{bmatrix} 0 & -185r & -80v \\ 185r & 0 & 30u \\ 80v & -30u & 0 \end{bmatrix} \qquad (۶)$$

(۷)

$$D(\vec{v}) = diag\{70 + 100|u|, 100 + 200|v|, 50 + 100|r|\}$$

با استفاده از این مقادیر و رابطه‌ی (۱) و نیز تعریف بردار $\vec{v}$ ، معادلات دینامیکی ربات نمونه در صفحه و در مختصات محلی بفرم روابط (۸) تبدیل می‌شود که از حالت ماتریسی خارج شده است:

(۸)

$$205\dot{u} - 265vr + (70 + 100|u|)u = X$$
$$265\dot{v} + 210ur + (100 + 200|v|)v = Y$$
$$80\dot{r} + 50uv + (50 + 100|r|)r = Z$$

این معادلات در مختصات متصل به جسم نوشته شده‌اند. جهت طراحی مسیر، چون مختصات کلی مبداء می‌باشد، این معادلات با استفاده از ماتریس انتقال (۲) به مختصات کلی منتقل می‌شوند. یعنی آنچه که بدست

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

می‌آید، همانند رابطه‌ی (۳) است. رابطه‌ی (۹)، معادلات حاصل از این انتقال در مختصات کلی را نشان می‌دهد.

(۹)

$$\dot{x} = u\cos\psi - v\sin\psi$$
$$\dot{u} = \frac{1}{205}(265\,(rv) - (70 + 100\,|u|)u + X)$$
$$\dot{y} = u\sin\psi + v\cos\psi$$
$$\dot{v} = \frac{1}{265}(215\,ru + (100 + 200\,|v|)v + Y)$$
$$\dot{\theta} = r$$
$$\dot{r} = \frac{1}{80}(50\,uv + (50 + 100\,|r|)r + M)$$

توجه شود که این معادلات علاوه بر انتقال به مختصات کلی، به فرم استاندارد شبیه به رابطه‌ی (۱۴) که در پی می‌آید تبدیل شده اند، یعنی ۳ معادله‌ی دیفرانسیل مرتبه دوم تبدیل به ۶ معادله‌ی دیفرانسیل مرتبه اول شده است و نیز سمت چپ معادلات، همگی مشتقات مرتبه اول هستند.

علاوه بر معادلات دینامیکی سیستم، قید وجود موانع نیز درنظر گرفته شده است. چنانچه در محیط حرکت ربات مانعی با مشخصات‌هندسی (x_{obs}, y_{obs}) موجود باشد، قید ناشی از وجود آن را بصورت زیر تعریف می‌کنیم:

$$(x - x_{obs})^2 + (y - y_{obs})^2 \geq a^2 \qquad (۱۰)$$

که درآن (x, y) موقعیت ربات در زمانهای مختلف و a کمترین فاصله‌ی درنظر گرفته شده بین ربات و مانع است.

حال اگر موانع متحرک باشند، این قید بفرم زیر تبدیل می‌شود:

$$(x - x(t)_{obs})^2 + (y - y(t)_{obs})^2 \geq a^2 \qquad (۱۱)$$

بهینه‌سازی با قیدهای نابرابری، عملی نسبتاً مشکل و زمان‌بر است. در نتیجه قید نابرابری (۱۱) تبدیل به قید برابری بصورت زیر می‌شود:

(۱۲)

$$\xi^2 = (x - x(t)_{obs})^2 + (y - y(t)_{obs})^2 - a^2$$

در این حالت مشاهده می‌شود که یک متغیر (ξ) به متغیرهای سیستم اضافه شده است.

۳- الگوریتم بهینه‌سازی:

طراحی مسیر برای ربات زیرسطحی در یک مانور حرکت صفحه‌ای به گونه‌ای که ربات از نقطه‌ی آغاز حرکت کرده و سپس با پرهیز از برخورد با موانع متحرک در محیط به‌سمت نقطه هدف میل کند، صورت می‌گیرد. تابع هدف بر اساس انرژی مصرفی توسط ربات معرفی می‌گردد و در ادامه تحقیق، حالاتی که چندین مانع متحرک در محیط باشد نیز مورد تحلیل واقع شده است. چون هدف، بهینه‌سازی در طراحی مسیر است، لازم است از یک روش بهینه‌سازی جهت حل طراحی مسیر بهینه استفاده شود. روش بهینه‌سازی انتخاب شده، برنامه‌ریزی دینامیک می‌باشد. این الگوریتم بدین شکل است که یک تابع توصیفی از سیستم و قیدهای آن با نام تابع همیلتونی[3] تشکیل می‌شود. فرم این تابع بصورت زیر است:

$$H = \vec{\lambda}^T \vec{F}(\vec{x}, \vec{U}) + OF \qquad (۱۳)$$

که در آن $\vec{F}(\vec{x}, \vec{U})$ تابعی است که سمت راست معادلات مشخصه‌ی سیستم است:

$$\dot{\vec{x}} = \vec{F}(\vec{x}, \vec{U}) \qquad (۱۴)$$

معادلات مشخصه‌ی سیستم در این مقاله، معادلات دینامیکی سیستم (۹) و نیز معادلات حاصل از قیدهای مسئله همچون قید وجود موانع و نیزمحدودیت عملگرها که در پی می‌آید، می‌باشد [8]. البته لازم است که معادلات مشخصه، تبدیل به فرم استاندارد همانند رابطه‌ی (۱۴) شوند. OF نیز تابعی است که باید در بهینه‌سازی، کمینه یا بیشینه شود. در این مسئله چون هدف، کمینه کردن انرژی است، داریم:

[3] Hamiltonian Function

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

$$OF = \int_0^T (X^2 + Y^2 + Z^2)dt \qquad (۱۵)$$

T مدت زمان انجام مأموریت توسط ربات است. رابطه‌ی (۱۵) اگرچه که میزان انرژی مصرفی را نشان نمی‌دهد، ولی معیاری از انرژی مصرفی ربات است. در الگوریتم بهینه‌سازی برنامه‌ریزی دینامیک، هدف کلی، بهینه کردن OF بوده و برای رسیدن به این هدف، لازم است که تابع همیلتونی (۱۳) بهینه شود [8]. جهت بهینه کردن تابع همیلتونی، باید مقادیر $\vec{\lambda}$ و $\vec{x}$ و نیز $\vec{U}$ به‌شکلی مناسب انتخاب شود. مقادیر $\vec{\lambda}$ و $\vec{x}$ را می‌توان با استفاده از روابط (۱۶) بدست آورد:

$$\begin{aligned} \dot{\lambda}_i &= -\frac{\partial H}{\partial x_i} \quad i = 1,2,..., n \\ \dot{x}_i &= \frac{\partial H}{\partial \lambda_i} \quad i = 1,2,..., n \end{aligned} \qquad (۱۶)$$

n تعداد متغیرهای سیستم است.

همچنین همانگونه که در ادامه خواهد آمد، بردار ورودیهای بهینه $\vec{U}$ را نیز می توان با توجه به نوع مسئله بدست آورد.

۴- حل مسئله‌ی طراحی مسیر برای ربات زیرسطحی

با استفاده از روش بهینه‌سازی ذکر شده، برای ربات نمونه، مسئله‌ی طراحی مسیر حل می‌شود. علاوه بر معادلات دینامیک سیستم، لازم است قید نابرابری مانع نیز لحاظ شود. حل مسئله‌ی بهینه سازی با استفاده از قیدهای نابرابری، همانگونه که ذکر شد،کاری بسیار مشکل و وقت‌گیر است. بنابراین قید تبدیل شده در رابطه‌ی (۱۲) به‌عنوان معیار قید وجود مانع درنظر گرفته می‌شود. حال باید از این قید یک معادله‌ی دیفرانسیل استخراج کرد تا به‌فرم استاندارد درآید. با مشتق‌گیری از طرفین رابطه‌ی(۱۲)، رابطه‌ی (۱۷) حاصل می‌شود:

$$\begin{aligned} \dot{\xi}\xi = (x - x(t)_{obs})(\dot{x} - \dot{x}(t)_{obs}) + \\ (y - y(t)_{obs})(\dot{y} - \dot{y}(t)_{obs}) \end{aligned} \qquad (۱۷)$$

مشتق‌گیری به‌این دلیل صورت می‌گیرد که قیدهای استاندارد در برنامه‌ریزی دینامیک، بصورت معادلات دیفرانسیل مرتبه‌ی اول می‌باشند. ولی استفاده از این قید بدون حضور ورودیها در آن، به بهینه سازی ناشی از این قید در الگوریتم بهینه‌سازی مربوطه، ترتیب اثر نخواهد داد [8]. با یک‌بار دیگر مشتق‌گیری از رابطه‌ی (۱۷) و جایگزینی مشتقات اول و دوم متغیرهای سیستم از رابطه‌ی (۹)، این قید اصلاح می‌شود. در نهایت، معادله‌ی دیفرانسیل مربوطه به معادلات سیستم اضافه می‌شود. این باعث خواهد شد که یک متغیر جدید دیگر نیز به عنوان ζ به‌صورت زیر به سیستم معادلات بهینه‌سازی افزوده شود :

$$\zeta = \dot{\xi} \qquad (۱۸)$$

رابطه‌ی (۱۸)، جهت تبدیل معادله‌ی دیفرانسیل به فرم استاندارد بدست آمده است. یک قید دیگر بصورت رابطه‌ی (۱۹) تعریف می‌شود:

$$\sigma^2 = (\theta + a)^2 \qquad (۱۹)$$

این قید بدین علت انتخاب شده است که ورودی‌های بهینه حاصل از روابط (۲۶) که در پی می‌آید، همگی هم تابعی از متغیرهای سیستم ($\vec{x}$) و هم تابعی از متغیرهای بهینه‌سازی ($\vec{\lambda}$) شوند. در اینجا، این موضوع در مورد ورودی Z صادق است که اگر رابطه‌ی (۱۹) استفاده نشود، این ورودی، ورودی بهینه‌ی لازم نخواهد بود. در این رابطه σ یک متغیر جدید و a یک عدد ثابت دلخواه است. با دو بار مشتق‌گیری، مانند رابطه‌ی (۱۷)، این قید اصلاح شده و تبدیل به معادله‌ی دیفرانسیل می‌شود(همانند آنچه که برای رابطه‌ی (۱۲) بدست آمد). حال چون محیط دینامیک است و باید همه‌ی قیدها به‌فرم استاندارد درآیند، لازم است که با تعریف متغیر زیر، این متغیر جایگزین زمان در معادلات سیستم شود:

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

$$\dot{\mu}=1,\mu(0)=0 \qquad (۲۰)$$

که در آن μ معادل زمان در سیستم معادلات می‌باشد. در حالت اول طراحی مسیر، یک مانع در محیط موجود است. مختصات مانع در این حالت مسئله که تابعی از زمان است، بصورت رابطه‌ی (۲۱) تعریف می‌شود:

$$\begin{aligned} x(\mu)_{obs} &= 2.1\mu \\ y(\mu)_{obs} &= 11-2.1\mu \end{aligned} \qquad (۲۱)$$

با جایگذاری این عبارات در معادله‌ی (۱۷) ، قید استاندارد ناشی از وجود مانع حاصل می‌شود. با درنظر گرفتن محدودیت عملگرها، تعداد متغیرهای سیستم افزایش می‌یابد.در این حالت:

$$|X|<b_1,|Y|<b_2,|Z|<b_3 \qquad (۲۲)$$

که در آن $\pm b_1$، $\pm b_2$ و نیز $\pm b_3$ بترتیب محدوده‌ی بالایی و پایینی نیروها (X و Y)و گشتاور (Z) را مشخص می‌کند. در اینجا جهت سادگی فقط ورودی X در مسئله محدود شده است:

$$|X|<b_1 \qquad (۲۳)$$

در این مسئله $b_1=700N$ درنظر گرفته شده است. همانند رابطه‌ی (۱۲)، این نامساوی نیز تبدیل به تساوی معادل می‌شود:

$$b_1-|X|=\alpha^2 \qquad (۲۴)$$

با مشتق‌گیری و تبدیل رابطه‌ی (۲۴) به معادله‌ی دیفرانسیل، روابط (۲۵) حاصل می‌شود:

$$\begin{aligned} \dot{X} &= X_2 \\ -\frac{XX_2}{|X|} &= 2\alpha\dot{\alpha} \end{aligned} \qquad (۲۵)$$

این روابط نیز به سیستم معادلات دیفرانسیل اضافه می‌شود. بدین شکل، دو متغیر دیگر به سیستم معادلات اضافه می‌شوند (α و X_2) . به ازای این دو متغیر، دو متغیر بهینه‌سازی متناظر نیز به سیستم اضافه خواهد شد. در حقیقت به ازای هر متغیر سیستم، یک متغیر بهینه‌سازی شکل خواهد گرفت.

مسئله‌ی طراحی مسیر بدین شکل حل می‌شود که با استفاده از تابع همیلتونی اصلاح شده و نیز با بدست آوردن ضرایب λ_i و x_i با استفاده از روابط (۱۶) ، در مجموع سیستم معادلات ۲۶ تایی حاصل می‌شود که ۱۳ تای آن مربوط به سیستم (معادلات حرکت (۹)، معادلات دیفرانسیل حاصل از روابط (۱۲)، (۱۸)، (۱۹) ،(۲۰) و(۲۵)) و ۱۳ تای آن نیز مربوط به متغیرهای بهینه‌سازی است. در نتیجه طبق رابطه‌ی (۱۶)، یک سیستم معادلات دیفرانسیل ۲۶ تایی غیرخطی باید حل شود. این معادلات، معادلات دیفرانسیل با شرایط مرزی بوده که شرایط اولیه و نهایی متغیرهای سیستم(و نه متغیرهای بهینه‌سازی) مشخص است. برخی از مقادیر اولیه و نهایی متغیرهای سیستم را می‌توان با استفاده از معادلات برابری و نیز با داشتن مقدار اولیه و نهایی موقعیت ربات، بدست آورد(مانند رابطه (۱۹) برای σ).با استفاده از حل عددی نیوتون-رافسون، این معادلات که یک مسئله‌ی مقدار مرزی است حل شده و خروجیهای متغیرهای سیستم($\vec{\eta}$) در زمانهای مختلف و جدای از مقدار اولیه و نهایی آنها بدست می‌آیند.

برای اینکه معادلات دیفرانسیل مربوطه حل شوند، لازم است که بجای ورودیها، مقادیر بهینه‌ی آنها که برحسب متغیرهای سیستم و متغیرهای بهینه‌سازی (و نه فقط بر حسب یکی از آنها) است، بدست آیند. در نتیجه با استفاده از معادلاتی که توسط روابط (۲۶) بدست می‌آیند، می‌توان ورودیهای بهینه را بدست آورد [9]:

$$\frac{\partial H}{\partial X}=0,\frac{\partial H}{\partial Y}=0,\frac{\partial H}{\partial Z}=0 \qquad (۲۶)$$

در نتیجه، معادلات بدست آمده از روابط (۱۶)، فقط تابعی از متغیرهای سیستم و متغیرهای بهینه‌سازی خواهند شد. همچنین همانگونه که ذکر شد، علت انتخاب رابطه‌ی (۱۹)، بدین جهت بود که ورودی Z تابعی از متغیرهای سیستم نیز باشد.

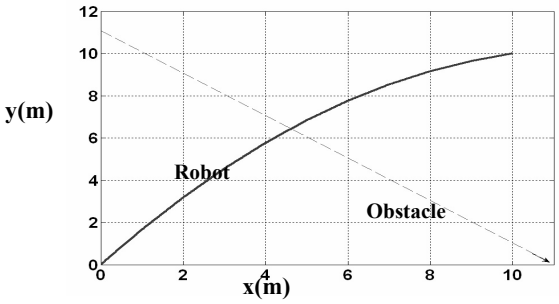
[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

۵- نتایج عددی

نتایج عددی بدست آمده با این فرض صورت گرفته است که شرایط ابتدایی و شرایط انتهایی بترتیب برابر است با:

$$\vec{\eta}(0) = [0\ 0\ 0]^T \qquad (۲۷)$$
$$\dot{\vec{\eta}}(0) = [0\ 0\ 0]^T$$
$$\vec{\eta}(10) = [10\ 10\ 0]^T \qquad (۲۸)$$
$$\dot{\vec{\eta}}(10) = [0\ 0\ 0]^T$$

در اینجا، مدت زمان فرآیند طراحی مسیر برای این مسئله، $T = 10$ ثانیه است.مختصات مانع نیز که تابعی از زمان است، همان معادلات (۲۱) هستند که برای حالت اول طراحی مسیر در نظر گرفته شده‌اند. در این حالت سرعت مطلق مانع برابر با $V_{obs} = 2.97\,m/s$ است. نتیجه پس از اجرای برنامه به‌صورت شکل (۲) نشان داده شده است.در این شکل، فقط راستای مسیرها مشخص‌اند. موقعیت ربات و موقعیت مانع در لحظات مختلف در جدول (۱) نشان داده شده است.



شکل ۲- راستای مسیرهای بدست آمده برای ربات و مانع در حالت اول طراحی مسیر

جدول ۱- موقعیت ربات و مانع در حالت اول طراحی مسیر

زمان(S)	مانع	ربات
۲/۵	(5.2,5.7)	(2.1,0.6)
۵	(10.5,0.5)	(5.3,2.2)
۷/۵	(15.7,–4.7)	(8.5,7.5)
۱۰	(21,–10)	(10,10)

مشخص است که مولفه‌های سرعت مانع، همان ضرایب μ هستند.

موقعیت مانع و ربات در جدول (۱) نشان داده شده است.

برای حالت اول طراحی مسیر می‌توان میزان مصرف انرژی را با استفاده از رابطه‌ی زیر محاسبه کرد:

$$E = \int_{x(t=0)}^{x(t=10)} X dx + \int_{y(t=0)}^{y(t=10)} Y dy + \int_{\theta(t=0)}^{\theta(t=10)} Z d\theta \qquad (۳۱)$$

برای این ربات، میزان انرژی مصرف شده با استفاده از رابطه‌ی (۳۱) برابر با ۹۵۳۸ ژول حاصل شده است.

مقادیر ورودیهای سیستم که در معادله‌ی (۳۱) جایگذاری شده‌اند، همان ورودیهای بهینه‌ای هستند که از روابط (۲۶) حاصل شده‌اند.

۶- طراحی سیستم کنترل

مسیر طراحی شده در قسمت ۵ ، مسیری است که توسط سیستم هدایت ربات تصمیم‌گیری می‌شود. حال جهت اجرای عمل تعقیب مسیر توسط ربات، باید یک سیستم کنترل خودکار برای ربات طراحی شود. این سیستم کنترل خودکار، برگرفته از روش اسلاتین و لی می‌باشد[5]. الگوریتم کلی این روش به صورتی است که در آن بردار متغیر $\vec{\eta}_d = [x_d, y_d, \theta_d]$ بعنوان ورودی مطلوب درنظر گرفته شده است. با تعریف $\tilde{\vec{\eta}} = \vec{\eta} - \vec{\eta}_d$ بعنوان خطای ردیابی، عملگر کنترلی باید بگونه‌ای طراحی شود که متغیر ردیابی زیر بسمت صفر همگرا شود:

$$\vec{s} = \dot{\tilde{\vec{\eta}}} + \lambda \tilde{\vec{\eta}} \qquad (۳۲)$$

که در آن λ یک ثابت مثبت بوده و بعنوان پهنای باند کنترل شناخته می‌شود. در این مسئله، $\lambda = 8.5$ درنظر گرفته شده است که با سعی و خطا حاصل شده است. مقادیر بیشتر نیز دقت کار را در تعقیب مسیر افزایش می‌دهد ولی در عمل بدلیل محدودیت عملگرها این کار عملی نیست. در نتیجه همگرایی $\vec{s}$ به‌سمت صفر دلالت بر این دارد که خطای ردیابی $\tilde{\vec{\eta}}$ بسمت

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

صفر میل می‌کند. معادله‌ی بالا جهت سادگی بصورت زیر نوشته می‌شود که بر حسب $\vec{\eta}_r$ بر طبق عبارت (۳۳) تعریف می‌شود:

$$\vec{s} = \dot{\vec{\eta}} - \dot{\vec{\eta}}_r \Rightarrow \quad \dot{\vec{\eta}}_r = \dot{\vec{\eta}}_d - \lambda\tilde{\vec{\eta}} \qquad (۳۳)$$

با استفاده از رابطه‌ی زیر می‌توان $\vec{\eta}_r$ را به مختصات متصل به جسم انتقال داد:

$$\dot{\vec{\eta}}_r = J(\vec{\eta})\vec{v}_r \qquad (۳۴)$$

در نتیجه بردارهای $\vec{v}_r$ و $\dot{\vec{v}}_r$ متصل به جسم را می‌توان بصورت رابطه‌ی (۳۵) نوشت:

$$\begin{aligned} \vec{v}_r &= J^{-1}(\vec{\eta})\dot{\vec{\eta}}_r \\ \dot{\vec{v}}_r &= J^{-1}(\vec{\eta})\left[\ddot{\vec{\eta}}_r - \dot{J}(\vec{\eta})J^{-1}(\vec{\eta})\dot{\vec{\eta}}_r\right] \end{aligned} \qquad (۳۵)$$

و در نهایت سیستم کنترل را بشکل زیر طراحی نمود:

$$\begin{aligned} \vec{\tau} = M\dot{\vec{v}}_r + C(\vec{v})\vec{v}_r + D(\vec{v})\vec{v}_r + g(\vec{\eta}) \\ - J^T(\vec{\eta})K_d\vec{s} \end{aligned} \qquad (۳۶)$$

که $\vec{v}_r$ و $\dot{\vec{v}}_r$ همان بردارهای تعریف شده در معادلات (۳۵) هستند. K_d نیز یک ماتریس مثبت بوده و بعنوان ماتریس پارامترهای طراحی شناخته می‌شود. $\vec{s}$ نیزمتغیر تعریف شده در معادله‌ی (۳۲) است. با بدست آمدن ورودی $\vec{\tau}$ از معادله‌ی (۳۶)، این مقدار وارد معادلات حرکت سیستم شده و پاسخ در هر لحظه بدست می‌آید. سیستم مدار بسته متشکل از کنترل‌کننده و سیستم دینامیکی با الگوریتم ذکر شده، در شکل (۳) نشان داده شده است. در اینجا ماتریس پارامترهای طراحی بصورت رابطه‌ی (۳۷) انتخاب می‌شود:

$$K_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (۳۷)$$

پاسخ سیستم کنترلی بشکل نمودار در شکلهای (۲۰) تا (۲۳) نشان داده شده است. این نمودارها، تعقیب مسیر ربات را در هر یک از راستاها و درجات آزادی سیستم و نیز در صفحه نشان می‌دهند. بازه‌ی زمانی در این نمودارها برابر با $\text{Time Step} = 0.014\text{s}$ انتخاب شده است. بنابراین مدت زمان کلی به ۷۰۰ بازه‌ی زمانی تقسیم شده که معادل مدت زمان طراحی و طی مسیر در ۱۰ ثانیه خواهد بود.

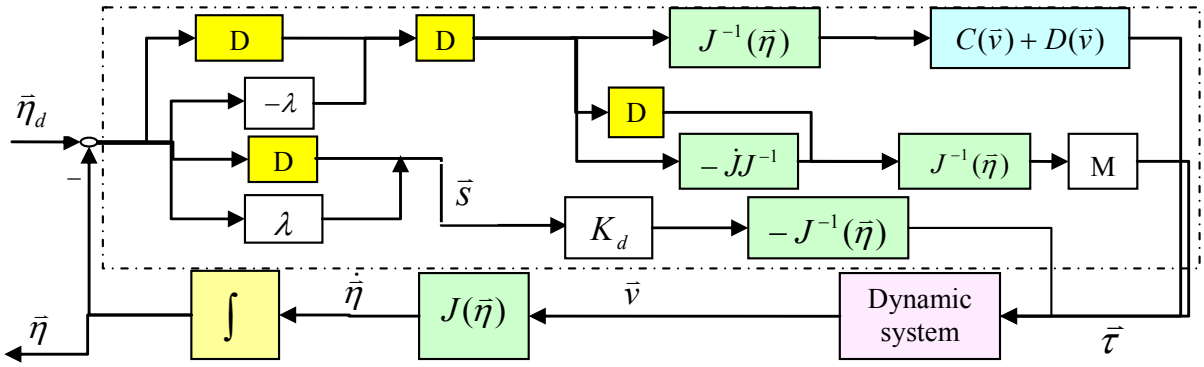
آنچه که در فرآیند سیستم کنترل به روش اسلاتین و لی دیده می‌شود این‌است که ابتدا دینامیک سیستم از مختصات کلی به مختصات محلی منتقل شده و پس از انجام یک سری عملیات ریاضی، در نهایت خروجی سیستم($\vec{v}$) با استفاده از ماتریس تبدیل مختصات، به مختصات کلی منتقل می‌شود. ورودیهای سیستم کنترلی نیز برحسب زمان در شکلهای (۸)، (۹) و (۱۰) نشان داده شده اند. علت جهش ورودیها در این شکلها این است که ورودی مطلوب $\vec{\eta}_d$ که یک سیگنال نامنظم است، به چهارده قسمت مساوی (برحسب زمان)تقسیم شده و هر قسمت با یک پاره خط شیبدار(ramp) تقریب زده شده است، این باعث می‌شود که در هنگام تغییر جهت مقدار ورودی مطلوب، بعلت تغییر ناگهانی گرادیان، ورودیها جهش ناگهانی داشته باشند تا مقدار مطلوب حاصل شود.

۷- طراحی مسیر در محیط‌های پیچیده‌تر

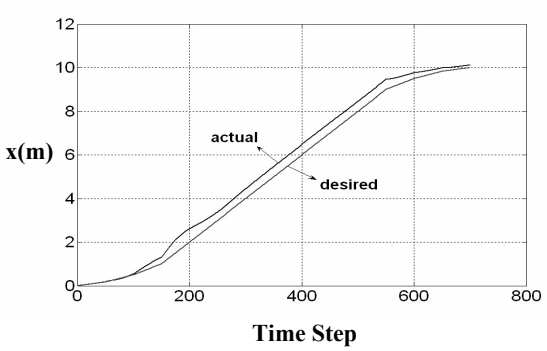
علاوه بر حالتی که در آن فقط یک مانع متحرک وجود داشت، دو حالت دیگر طراحی مسیر انجام شده است. در حالتهای دوم و سوم نیز طراحی مسیر در مدت زمان ۱۰ ثانیه انجام گرفته است. در این دو حالت، جهت ساده‌تر شدن مسئله، محدودیت عملگرها درنظر گرفته نشده و نیز به ازای وجود هر مانع، یک قید مانند رابطه‌ی (۱۰) وجود خواهد داشت که با تبدیل آن به رابطه‌ی (۱۲) و نیز تبدیل آن به قید استاندارد (همانند روابط (۱۷) و (۱۸))، و در نهایت اضافه کردن آن به تابع همیلتونی، با استفاده از روابط (۱۶)، معادلات دیفرانسیل مربوط به بهینه‌سازی حاصل شده و نیز ورودیهای بهینه با استفاده از همان روابط(۲۶) بدست می‌آیند. در این حالات، تابع همیلتونی برابر خواهد بود با:

$$\begin{aligned} H = \vec{\lambda}^T\vec{F}(\vec{x},\vec{U}) + \vec{\lambda}_1\vec{F}_1(\vec{x},\vec{U}) \\ + \vec{\lambda}_2\vec{F}_2(\vec{x},\vec{U}) + \ldots \end{aligned} \qquad (۳۸)$$

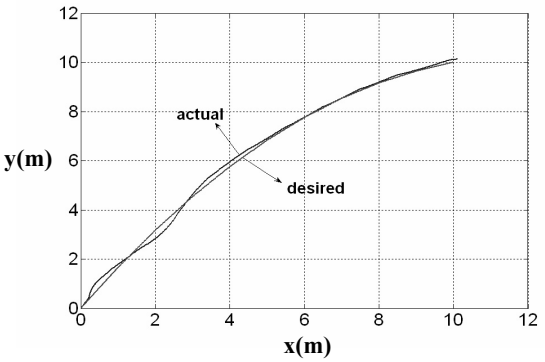
[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]



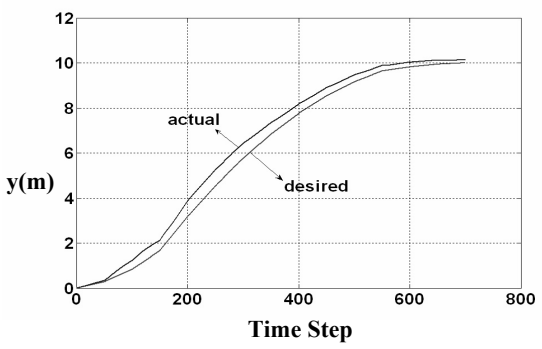
شکل ۳- سیستم کنترلی طراحی شده به‌روش اسلاتین و لی جهت تعقیب مسیر



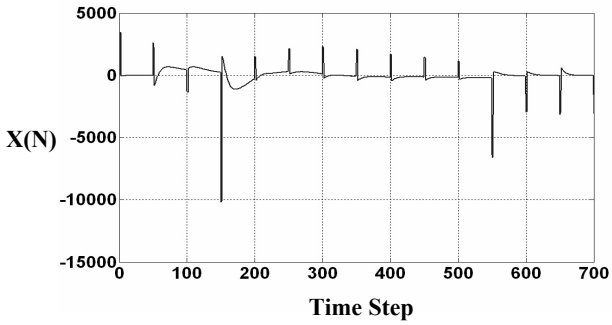
شکل ۴- تعقیب مسیر x بر حسب بازه‌ی زمانی



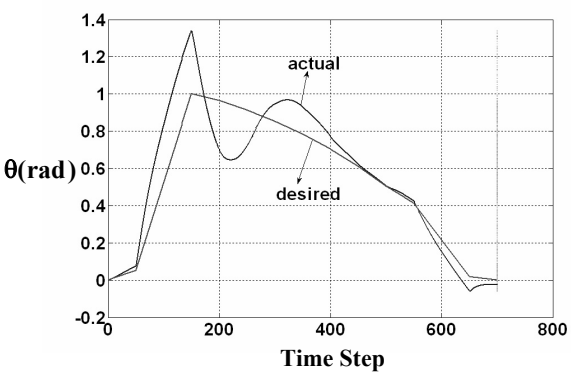
شکل ۷- تعقیب مسیر در صفحه‌xy



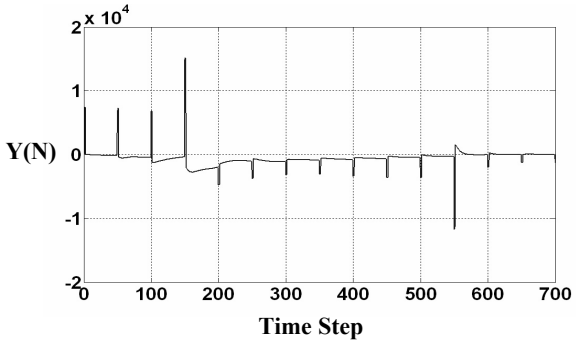
شکل ۵- تعقیب مسیر y بر حسب بازه‌ی زمانی



شکل ۸- ورودی کنترلی نیرو در راستای حرکت پیشروی به سیستم دینامیکی برحسب بازه‌ی زمانی

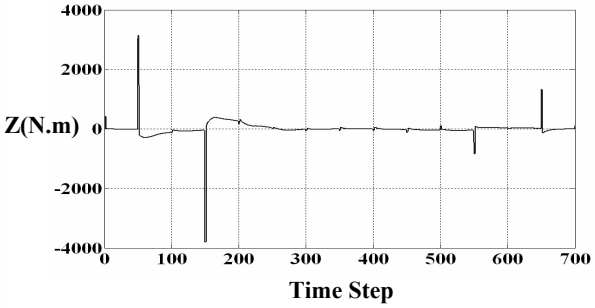


شکل ۶- تعقیب مسیر زاویه‌ای بر حسب بازه‌ی زمانی



شکل ۹- ورودی کنترلی نیرو در راستای حرکت عرضی به سیستم دینامیکی برحسب بازه‌ی زمانی

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]



شکل ۱۰- ورودی کنترلی گشتاور در حرکت گردشی به سیستم دینامیکی برحسب بازه‌ی زمان

که در آن $\vec{F}$، سمت راست معادلات دیفرانسیل (۸) و $\vec{F}_i$ نیز سیستم معادلات حاصل شده به ازای وجود هر قید(در اینجا قیدهای ناشی از وجود موانع) می‌باشد. $\vec{\lambda}_i$ ها نیز بردار متغیرهای بهینه‌سازی به ازای متغیرها و معادلات اضافه شده‌ی ناشی از قیدها می‌باشند[8]. در حالت دوم، دو مانع با معادلات حرکتی زیر در محیط انتخاب شده‌اند:

$$\begin{cases} (x(t)_{obs(1)}, y(t)_{obs(1)}) = (7-2t^2, 3.6t^2) \\ (x(t)_{obs(1)}, y(t)_{obs(1)}) = \\ \quad (8+2\cos(60t), 8+2\sin(60t)) \end{cases} \quad (۳۹)$$

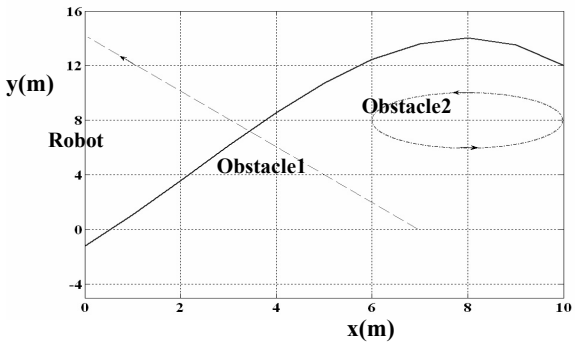
شرایط اولیه و نهایی ربات بدین صورت انتخاب شده است:

$$x(0)=0;\ y(0)=-1;\ u(0)=0$$
$$v(0)=0;\ \theta(0)=0;\ r(0)=0$$
$$x(10)=10;\ y(10)=12;\ u(10)=0$$
$$v(10)=0;\ \theta(10)=0;\ r(10)=0$$

حداقل فواصل موانع اول و دوم تا ربات به‌ترتیب ۳ و ۴ متر هستند. نتیجه‌ی نهایی امتداد حرکت ربات و موانع بدون برخورد و با تابع هدف (۱۵) بصورت نمودار در شکل (۱۱) آمده است. موقعیت موانع و ربات نیز در زمانهای مختلف در جدول (۲) آمده است.

در حالت دوم، انرژی مصرفی ربات با استفاده از رابطه (۳۱) برابر با ۱۳۱۲۳ ژول حاصل شده است.

در حالت سوم این‌بار سه مانع متحرک در محیط قرار داده می‌شوند.



شکل ۱۱- راستای مسیرهای موانع و ربات در حالت دوم

جدول ۲- موقعیت ربات و موانع در حالت دوم

زمان(S)	مانع اول	مانع دوم	ربات
۲/۵	(−5.5,22.5)	(9.4,6.57)	(1.5,2)
۵	(−43,90)	(7.96,6)	(3.7,7.8)
۷/۵	(−105.5,202.5)	(6.53,6.63)	(7.5,13.5)
۱۰	(−193,360)	(6,8.08)	(10,12)

معادلات حرکت موانع به‌ترتیب به‌صورت روابط (۴۰) می‌باشند:

$$\begin{cases} (x(t)_{obs(1)}, y(t)_{obs(1)}) = (0.5t^2, 8-2t) \\ (x(t)_{obs(1)}, y(t)_{obs(1)}) = (5, 6\sin(100t)) \\ (x(t)_{obs(3)}, y(t)_{obs(3)}) = (9, -6+1.4t) \end{cases} \quad (۴۰)$$

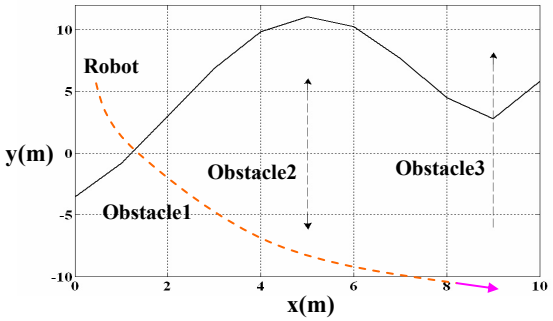
شرایط اولیه و نهایی ربات به‌صورت زیر انتخاب شده است:

$$x(0)=0;\ y(0)=-3.5;\ u(0)=0$$
$$v(0)=0;\ \theta(0)=0;\ r(0)=0$$
$$x(10)=10;\ y(10)=6;\ u(10)=0$$
$$v(10)=0;\ \theta(10)=0;\ r(10)=0$$

حداقل فواصل ربات تا موانع اول، دوم و سوم به‌ترتیب برابر با ۶ ، ۴ و ۳ متر می‌باشند. نتیجه ی نهایی امتداد حرکت ربات و موانع بدون برخورد و با تابع هدف بصورت نمودار در شکل (۱۲) نشان داده شده است. همچنین موقعیت موانع و ربات بر حسب زمان در جدول (۳) نشان داده شده است. در حالت سوم نیز

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

انرژی مصرفی ربات با استفاده از همان رابطه (۳۱) برابر با ۱۶۰۳۹ ژول حاصل شده است.



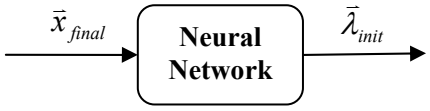
شکل ۱۲- راستای مسیرهای ربات و موانع در حالت سوم

جدول ۳- موقعیت ربات و موانع در حالت سوم

زمان(S)	مانع اول	مانع دوم	مانع سوم	ربات
۲/۵	(0.43,3)	(5,–5.8)	(9,–2.5)	(1,–0.75)
۵	(1.5,–2)	(5,–2.8)	(9,1)	(3.9,9.8)
۷/۵	(5.82,–7)	(5,4.47)	(9,4.5)	(8.3,3.87)
۱۰	(20.13,–12)	(5,4.96)	(9,8)	(10,6)

۸- طراحی مسیر با استفاده از حل تبدیل هوشمند

در حالتی که مسئله طراحی مسیر طولانی‌تر شده و ربات بخواهد مأموریت طولانی‌تری را انجام دهد، روشهای کلاسیک معمول، قادر به حل معادلات دیفرانسیل با شرایط مقدار مرزی نیستند. در اینجا، با استفاده از روش حل تبدیل هوشمند، این معادلات حل شده و حل مسئله‌ی طراحی مسیر کامل می‌شود. این روش بدین شکل است که ابتدا برای مسئله موردنظر، مقادیر اولیه‌ی دلخواه به متغیرهای بهینه‌سازی داده شده و بجای اینکه معادلات با شرایط مرزی حل شوند، معادلات با شرایط اولیه حل می‌شوند. در این حالتها، روش حل بجای نیوتون رافسون، رانگ کوتا خواهد بود. سپس حالتهای نهایی سیستم به ازای هر مقدار اولیه دلخواه، جمع آوری شده و بعنوان ورودیهای شبکه عصبی انتخاب می‌شود. خروجیهای شبکه نیز، مقادیر اولیه‌ی متغیرهای بهینه سازی خواهند بود. این ارتباط در شکل (۱۳) نشان داده شده است. در این شکل، $\vec{\lambda}_{init}$ بردار مربوط به مقدار اولیه متغیرهای بهینه‌سازی و $\vec{x}_{final}$ مقدار



شکل ۱۳- رابطه بین ورودیها و خروجیها در شبکه عصبی

نهایی مربوط به متغیرهای سیستم است. با آموزش این شبکه، میتوان مقدار اولیه متغیرهای بهینه‌سازی متناظر با مقدار نهایی متغیرهای سیستم را بدست آورد. در نتیجه، بجای حل مسئله مقدار مرزی، مسئله مقدار اولیه‌ای حل می‌شود که تقریباً سیستم را به موقعیت نهایی می‌برد. با اینکار، نه تنها مدت زمان حل معادلات دیفرانسیل کاهش می‌یابد، بلکه جواب بصورت قطعی همگرا هم می‌شود. هرچه آموزش شبکه بهتر باشد، انحراف سیستم از مقدار نهایی کمتر خواهد بود. جزییات بیشتر این روش در مرجع [10] موجود است.

در اینجا، با توجه به شرایط مسئله، می‌توان عمل آموزش را انجام داد. در حالت ششم، شرایط مرزی ربات چنین درنظر گرفته شده است:

$$\vec{x}_{init} = [1,1,0]$$
$$\vec{x}_{final} = [10,13,0]$$

مانند آنچه که در روش حل تبدیل هوشمند توضیح داده شد، با دادن مقادیر اولیه دلخواه به متغیرهای بهینه‌سازی، مقادیر نهایی بدست آمده متناظر از سیستم را با مقادیر اولیه متغیرهای بهینه سازی، بترتیب بعنوان ورودی و خروجی شبکه آموزش‌دیده درنظر می‌گیریم. در حالت چهارم، دو مانع که یکی ثابت و دیگری متحرک است، در محیط درنظر گرفته می‌شوند. در حالت کلی، طبق آنچه که در قسمتهای ۴ و ۵ توضیح داده شد، و درنظر نگرفتن محدودیت عملگرها جهت سادگی، تعداد کل متغیرهای سیستم به عدد ۱۳ می‌رسد که تعداد متغیرهای بهینه‌سازی متناظر نیز طبق مطالب ذکر شده، همان ۱۳ عدد

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

خواهد بود. در نتیجه شبکه عصبی آموزش داده شده دارای ۱۳ ورودی و ۱۳ خروجی است. ساختار کلی این شبکه در شکل (۱۴) نشان داده شده است.

برای لایه میانی، توابع بصورت زیر درنظر گرفته شده‌اند:

$$z_j = \frac{1}{1+\exp(-X_j)}$$
$$X_j = \sum_{i=1}^{4} w_{ij} x_i \qquad (۴۱)$$

که در این حالت داریم:

$$W = \{w_{ij}\}$$

برای لایه خارجی شبکه نیز، توابع بصورت زیر درنظر گرفته شده‌اند:

$$y_k = \frac{2}{1+\exp(-Z_k)} - 1$$
$$Z_k = \sum_{j=1}^{6} v_{jk} z_j \qquad (۴۲)$$

که در آن:

$$V = \{v_{jk}\}$$

مشخصات مانع ثابت بصورت زیر است:

$$(x_{obs1}, y_{obs1}) = (6,8) \qquad (۴۳)$$

مشخصات مانع متحرک نیز بصورت زیر است:

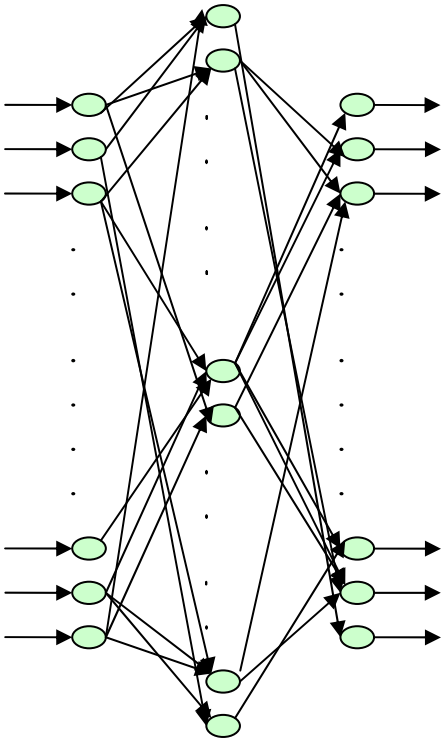
$$(x(t)_{obs2}, y(t)_{obs2}) = (5-2t, 4.25t) \qquad (۴۴)$$

ورودیهای آموزش به شبکه در جدول (۴) آمده است. خروجیهای آموزش نیز در نمودارهای بدست آمده از آموزش(خروجی مطلوب) نشان داده شده است.

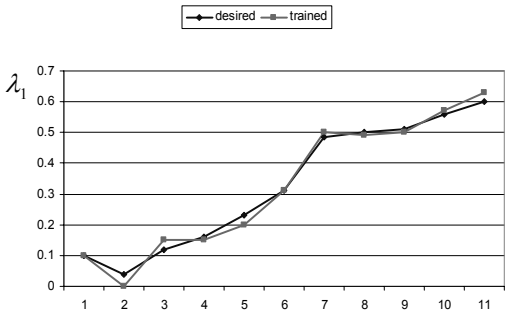
با آموزش شبکه، برای خروجی شبکه (مقادیر اولیه متغیرهای بهینه‌سازی) نمودارهای (۱۵) تا (۲۷) حاصل می‌شود.

حال درنظر است ربات به نقطه‌ی نهایی مطلوب منتقل گردد. نقطه نهایی مطلوب برابر است با :

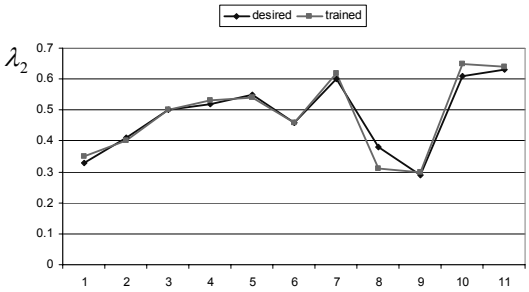
$$\vec{\eta} = [10,13,0]$$



شکل ۱۴- ساختار شبکه عصبی با ۱۷ نورون میانی و ۱۳ نورون ورودی و خروجی



شکل ۱۵- مقدار اولیه مطلوب λ_1 در مقایسه با مقدار بدست آمده از آموزش

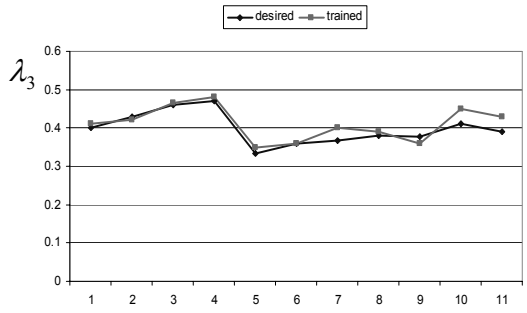


شکل ۱۶- مقدار اولیه مطلوب λ_2 در مقایسه با مقدار بدست آمده از آموزش

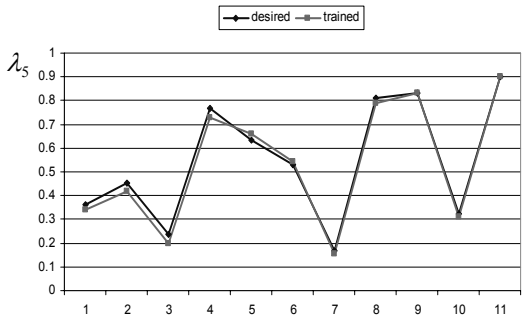
[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

جدول ۴- مقادیر نهایی سیستم بعنوان الگوهای ورودی به شبکه

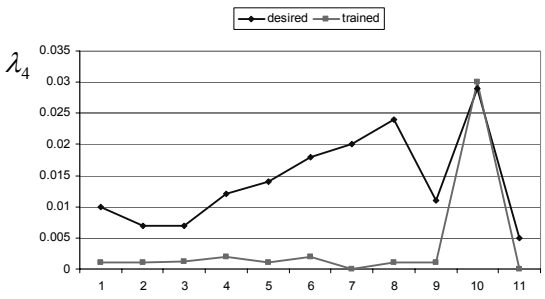
شماره الگو ⇐	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱
x_1	۱۰/۱	۹/۷	۹/۳	۱۱	۱۰/۶	۹/۴	۸/۷	۱۰/۴	۱۰/۳	۹/۸	۱۱/۱
x_2	۰/۳	۰/۱	۰/۱۲	-۰/۳۱	۰/۲۱	۰/۴۲	-۰/۰۷	۰/۰۲	۰/۳۱	-۰/۱۱	۰/۲۵
x_3	۱۳/۳	۱۲/۷	۱۲/۳	۱۳/۱	۱۱/۸	۱۲/۹	۱۳	۱۲/۱	۱۲/۸	۱۳/۳	۱۲/۵
x_4	۰/۳۱	۰/۴	-۰/۱	۰/۰۱	۰/۲	-۰/۲	۰/۰۳	۰	۰/۱۱	۰/۰۵	۰/۱۴
x_5	۰/۰۱	۰/۰۲	-۰/۰۲	۰/۰۳	۰/۰۳	۰/۰۹	-۰/۰۵	۰/۱	۰/۰۹	۰/۱	۰/۰۷
x_6	۰/۱	-۰/۱۱	۰/۰۸	۰/۰۸	-۰/۰۶	-۰/۰۲	۰/۰۵	۰	۰/۰۳	۰/۰۴	-۰/۰۵
x_7	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰
x_8	۱/۰۱	۱/۰۲	۰/۹۸	۱/۰۱	۱/۲	۰/۸	۱/۰۳	۱	۱/۱۱	۱/۰۵	۱/۱۴
x_9	۰/۱	-۰/۱۱	۰/۰۸	۰/۰۸	-۰/۰۶	-۰/۰۲	۰/۰۵	۰	۰/۰۳	۰/۰۴	-۰/۰۵
x_{10}	۸/۶	۱۲/۵	۶/۱	۹/۹	۱۰/۱	۸/۲	۷/۳	۸/۶	۴/۵	۸/۸	۱۱/۱
x_{11}	۵/۱۲	۵/۲۹	۶/۱۰	۵/۷۴	۶/۹۱	۶/۶	۵/۸	۵/۲۱	۶/۳۱	۵/۴۵	۶/۰۸
x_{12}	۲۱/۴	۲۵/۵	۲۴/۳	۲۴/۲	۲۱/۶	۲۲/۸	۲۳/۲	۲۳/۳	۲۴/۹	۲۵/۶	۲۳/۸
x_{13}	-۴/۴	-۴/۲	-۳/۹	-۳/۸	-۳/۶	-۴/۱	-۴/۹	-۳/۱	-۳/۶	-۴	-۳/۸



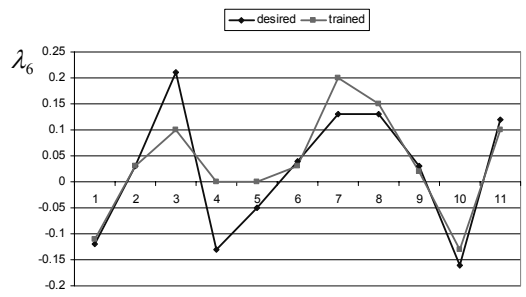
شکل ۱۷- مقدار اولیه مطلوب λ_3 در مقایسه با مقدار بدست آمده از آموزش



شکل ۱۹- مقدار اولیه مطلوب λ_5 در مقایسه با مقدار بدست آمده از آموزش

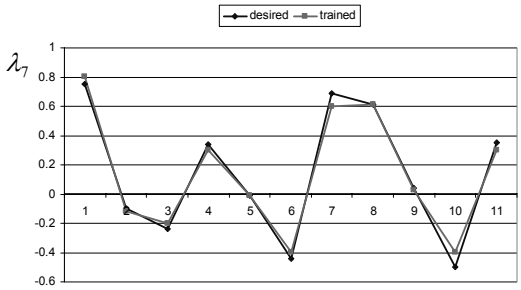


شکل ۱۸- مقدار اولیه مطلوب λ_4 در مقایسه با مقدار بدست آمده از آموزش

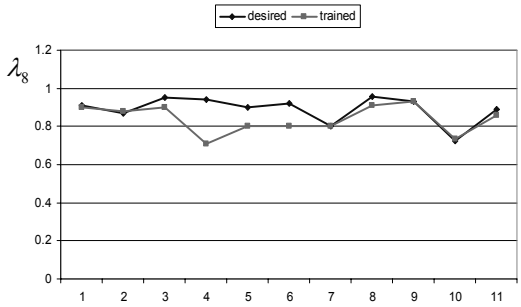


شکل ۲۰- مقدار اولیه مطلوب λ_6 در مقایسه با مقدار بدست آمده از آموزش

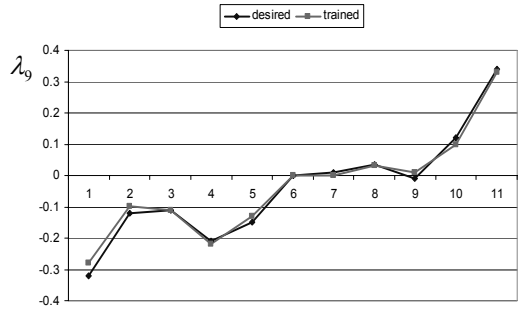
[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]



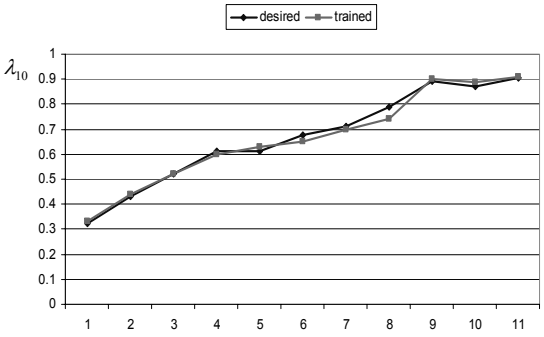
شکل ۲۱- مقدار اولیه مطلوب λ_7 در مقایسه با مقدار بدست آمده از آموزش



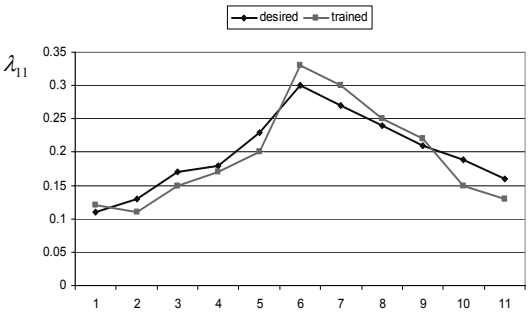
شکل ۲۲- مقدار اولیه مطلوب λ_8 در مقایسه با مقدار بدست آمده از آموزش



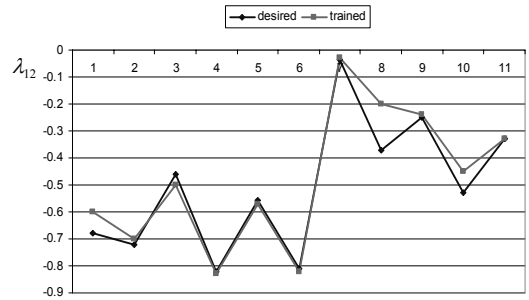
شکل ۲۳- مقدار اولیه مطلوب λ_9 در مقایسه با مقدار بدست آمده از آموزش



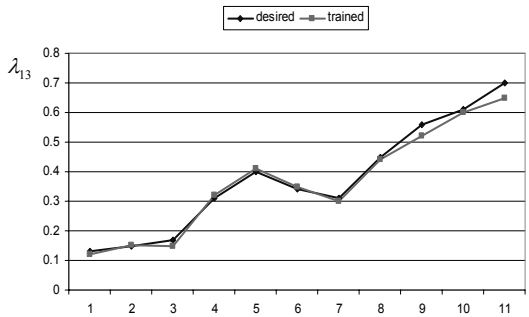
شکل ۲۴- مقدار اولیه مطلوب λ_{10} در مقایسه با مقدار بدست آمده از آموزش



شکل ۲۵- مقدار اولیه مطلوب λ_{11} در مقایسه با مقدار بدست آمده از آموزش



شکل ۲۶- مقدار اولیه مطلوب λ_{12} در مقایسه با مقدار بدست آمده از آموزش



شکل ۲۷- مقدار اولیه مطلوب λ_{13} در مقایسه با مقدار بدست آمده از آموزش

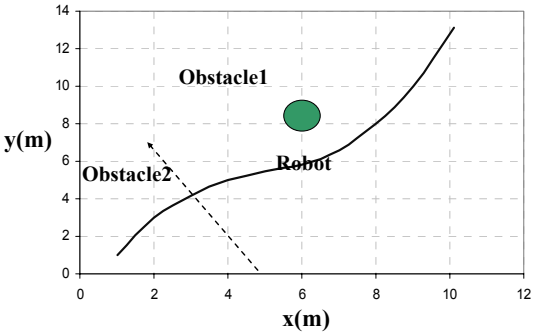
با دادن این مقادیر و سایر متغیرهای سیستم که از مشخصات و قیود مسئله بدست می‌آیند (۱۳ عدد) به شبکه عصبی آموزش داده شده، مقادیر اولیه حاصل شده از متغیرهای بهینه‌سازی بدست می‌آید. این مقادیر در جدول(۵) آمده‌اند.

حال با استفاده از این مقادیر و مقادیر اولیه متغیرهای سیستم، این بار سیستم معادلات دیفرانسیل با شرایط اولیه‌ای موجود است که با حل آن، سیستم تقریباً به نقطه نهایی (مقصد) می‌رسد. شکل (۲۸) راستای موقعیت ربات و موانع را در صفحه نشان می‌دهد.

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

جدول ۵- مقدار اولیه متغیرهای بهینه‌سازی بدست آمده از شبکه عصبی آموزش داده شده

متغیرهای بهینه سازی	مقدار اولیه	متغیرهای بهینه سازی	مقدار اولیه
λ_1	۰/۵۳	λ_8	۰/۹
λ_2	۰/۴۱	λ_9	-۰/۱۳
λ_3	۰/۳۹	λ_{10}	۰/۷۱
λ_4	۰/۰۰۱	λ_{11}	۰/۲۶
λ_5	۰/۶۴	λ_{12}	-۰/۳۴
λ_6	-۰/۰۵	λ_{13}	۰/۳۵
λ_7	۰/۳۲	———	———



شکل ۲۸- موقعیت موانع و ربات در حالت ششم طراحی مسیر

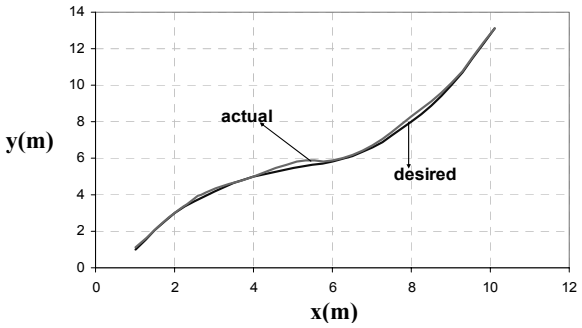
در این حالت نیز موقعیت ربات و موانع درمراحل زمانی مختلف در جدول (۶) نشان داده شده است. در این حالت، انرژی مصرفی ربات برابر با ۱۲۷۷۳ ژول محاسبه شده است.

جدول ۶- موقعیت ربات و موانع در حالت ششم

زمان(S)	مانع	مانع دوم	ربات
۲/۵	(6,8)	(2.5,5.1)	(3,4)
۵	(6,8)	(0,10.5)	(5.8,5.7)
۷/۵	(6,8)	(-2.5,15.75)	(8.3,8.9)
۱۰	(6,8)	(-5,21)	(10.2,13.1)

موضوعی که در اینجا جالب توجه است، اینست که مدت زمان حل معادلات دیفرانسیل بمراتب کاهش یافته و در این حالت، ۱/۵ ثانیه زمان برای حل لازم است. در نتیجه از این روش می‌توان برای حل مسئله طراحی مسیر زمان‌زنده (Real Time) استفاده کرد. در عوض، سیستم تا حدی از نقطه‌ی نهایی انحراف داردکه همانگونه که ذکر شد، میزان این انحراف وابسته به نوع شبکه‌ی عصبی و نوع آموزش آن است.

در این حالت، بخشی از مسئله با استفاده از روش کلاسیک و بخش دیگر با استفاده از روش هوشمند(شبکه‌های عصبی) حل شده است. به همین علت به این روش، حل تبدیل هوشمند گفته می‌شود. برای این متحرک می‌توان سیستم کنترلی نیز که در بخش (۶) طراحی شد، درنظر گرفت. با دادن ورودی به سیستم کنترلی، عمل تعقیب مسیر در حالت ششم نیز صورت گرفته است. نمودار مربوط به تعقیب مسیر، در شکل (۲۹) آمده است.



شکل ۲۹- تعقیب مسیر مطلوب طراحی شده توسط سیستم کنترل در حالت ششم

۹- نتیجه‌گیری

بعنوان نتیجه‌گیری کلی، می‌توان گفت که با اصلاح تابع همیلتونی و با استفاده از برنامه‌ریزی دینامیک، برای ربات‌های زیرسطحی در محیط‌های با موانع متحرک می‌توان طراحی مسیر بهینه انجام داد. همچنین باید ورودی‌های بهینه بگونه‌ای انتخاب شوند که تابعی از متغیرهای بهینه‌سازی و متغیرهای سیستم باشند. در غیر این صورت، ورودی بهینه یک ورودی مناسب نخواهد بود و امکان دارد که پاسخ بهینه برای ربات بدست نیاید. در مورد سیستم کنترلی نیز می‌توان نتیجه گرفت که چون مسیر بهینه‌ی بدست آمده، معمولاً یک مسیر هموار است، سیستم کنترل انتخابی جهت تعقیب مسیر، یک سیستم مناسب می‌باشد. همچنین، استفاده از حل تبدیل

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]

هوشمند راهکاری است برای کاهش مضاعف زمان حل معادلات دیفرانسیل و تبدیل آنها از یک مسئله‌ی مقدار مرزی به یک مقدار اولیه. در نتیجه از آن می‌توان جهت طراحی مسیر زمان‌زنده ربات استفاده کرد. در عوض، سیستم کمی از مقدار نهایی مطلوب انحراف دارد که وابسته به نوع شبکه عصبی انتخابی و آموزش آن است.

۱۰- نمادها

θ	زاویه
$\vec{\eta}$	بردارموقعیت در مختصات کلی
J	ماتریس انتقال
$\vec{v}$	بردار سرعت در مختصات محلی
C	ماتریس اثرات غیرمستقیم
D	ماتریس اثر مستقیم
M	ماتریس جرم
$\vec{\tau}$	بردار ورودیها
(x_{obs}, y_{obs})	موقعیت مانع
OF	تابع هدف در همیلتونی
$\vec{\lambda}$	بردار متغیرهای بهینه سازی
H	همیلتونی
ξ	متغیر کمکی قید حرکت مانع
ζ	متغیر کمکی تبدیل معادله
σ	متغیر کمکی جهت بهینه کردن ورودی
μ	متغیر جایگزین شده بجای زمان
t	زمان
$\vec{U} = \vec{\tau}$	بردار ورودیها
$\vec{\eta}_d$	موقعیت مطلوب
λ	پهنای باند(در سیستم کنترلی)
K_d	ماتریس پارامترهای طراحی
$\vec{x}$	بردار متغیرهای سیستم
E	انرژی مصرفی ربات در طراحی مسیر
W	ماتریس وزن برای لایه میانی
Z	ماتریس وزن برای لایه بیرونی

۱۱- مراجع

1-S.Waydo & Richard M.Murray, "Vehilcle Motion Planning Using Stream Funtions", IEEE International Conference on Robotics & Automation, Taipei,Taiwan, September 2003.

2-M.I.Riberio & P. Lima, "Motion Planning" Institute of Systems and Robotics ,Lisbon, Portugal, 2002, research paper.

3-J.Rief&M.Sharir,"Motion Planning in The Presence of Moving Obstacles", Tel Aviv & New York Universities, 1994, Journal of Association for Computing Machinery, Vol.41, pp 764-790.

4-Kiku Fujimura,"Time minimal Paths amidst Moving Obstacles in three Dimensions", International Journal on Theoretical Computer Science, Vol 270, January 2002, pp 421-440.

5-T.I.Fossen, "Guidance and Control of Ocean Vehicles", University of Trondheim, Norway, Copyright 1994 by John Wiley and Sons Ltd.

6-L.Lapierre, D.Soetanto, A.Pascoal , " Nonlinear Path Following with Applications to the Control of Autonomous Underwater Vehicles", Institute for Systems and Robotics and Department of Electrical Engineering and Computers, Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii, USA, December 2003.

7-T.I.Fossen, J.G.Balchen, "The NEROV Autonomous Underwater Vehicle", Department of Engineering Cybernetics, The Norwegian Institute of Technology, research report, 1995.

8-Byron S.Gottfried & Joel Weisman, "Introduction to Optimization Theory", Department of Industrial Engineering, University of Pittsburgh, Department of Chemical and Nuclear Engineering, University of Cincinnaty, Copyright 1973 by Prentice-Hall.

9-Morton.M.Denn,"Optimization By variational Methods", Department of Chemical Engineering, University of Delaware, Copyright 1969 by Mc Graw-Hill.

10-A.Reza Babakhani, Hassan Sayyaadi, "Application of Neural Network to Find Initial State of Optimization Parameters", ASME, IMEC2006 International Mechanical Engineering Congress and Exposition, Nov. 2006, Chicago, Illinois, USA.

[Downloaded from marine-eng.ir on 2026-11-04]
[DOR: 20.1001.1.17357608.1388.5.9.2.0]