هدایت و کنترل خودمختار ملاقات مداری با شش درجه آزادی با استفاده از فرایادگیری تقویتی و شبکه‌های ترنسفورمر

تاریخ دریافت: ۱۴۰۳/۰۹/۲۳
تاریخ پذیرش: ۱۴۰۴/۰۲/۱۹

مهرداد محسنی[۱]، ایمان محمدزمان[۲]
۱- دانشجو دکتری، مجتمع دانشگاهی برق و سایبرنتیک، دانشگاه صنعتی مالک‌اشتر
۲-دانشیار، مجتمع دانشگاهی برق و سایبرنتیک، دانشگاه صنعتی مالک‌اشتر،mohammadzaman@mut.ac.ir

چکیده

افزایش تعداد ماهواره‌ها در مدار پایین زمین، خطر برخورد بین اجسام فضایی را به‌شدت افزایش داده است. مأموریت‌های سرویس‌دهی و حذف زباله‌های فضایی می‌توانند با افزایش عمر ماهواره‌ها و پاکسازی مدارها، این تهدید را کاهش دهند. در این پژوهش، یک رویکرد نوین برای هدایت و کنترل فضاپیما در سناریوهای ملاقات مداری با شش درجه آزادی ارائه شده است که مبتنی بر یادگیری تقویتی فرا-آموزشی و شبکه‌های ترنسفورمر است. این مدل با کمک شبکه‌های ترنسفورمر، امکان یادگیری روابط پیچیده زمانی و استنباط اطلاعات پنهان از محیط را برای فضاپیمای تعقیب‌کننده فراهم می‌کند. الگوریتم بهینه‌سازی سیاست مجاورتی (PPO) که برای آموزش مدل به‌کار گرفته شده، در کنترل پیوسته عملکرد بالایی دارد. نتایج شبیه‌سازی‌ها در محیط مجازی نشان می‌دهند که این رویکرد از لحاظ دقت و پایداری بر معماری‌های سنتی مانند LSTM برتری دارد. از سوی دیگر تعداد پارامترهای شبکه، خود چالشی مهم در پیاده سازی بر روی سخت افزارها هست که روش پیشنهادی با کاهش محسوس در تعداد پارامترهای شبکه در کنار افزایش انطباق‌پذیری و بهبود دقت در شرایط متغیر محیطی کمک می کند. این رویکرد می‌تواند به‌عنوان راهکاری مؤثر برای تسهیل مأموریت‌های سرویس‌دهی و مدیریت زباله‌های فضایی مورد استفاده قرار گیرد.

واژه‌های کلیدی: *ملاقات مداری، فرایادگیری تقویتی، ترنسفورمر، سرویس مداری*

Autonomous six-degree-of-freedom orbital rendezvous guidance and control using meta-reinforcement learning and transformer networks

Mehrdad Mohseni[1] , Iman Mohammad Zaman [2]

1. PhD Student, Faculty of Electrical and Cybernetic Engineering, Malek Ashtar University of Technology, Iran
2. Associate Professor, Faculty of Electrical and Cybernetic Engineering, Malek Ashtar University of Technology, Iran. mohammadzaman@mut.ac.ir

Abstract

The increasing number of satellites in low Earth orbit has significantly heightened the risk of collisions between space objects. Servicing and debris removal missions offer a viable solution by extending satellite lifespans and clearing orbital pathways. This research presents an innovative approach for spacecraft guidance and control in six degrees-of-freedom orbital rendezvous scenarios, employing meta-reinforcement learning and transformer networks. Leveraging transformer networks, this model enables the chaser spacecraft to learn complex temporal relationships and infer hidden information from the environment. The Proximal Policy Optimization (PPO) algorithm, utilized for model training, demonstrates superior performance in continuous control tasks. Simulation results in a virtual environment indicate that this approach outperforms traditional architectures like LSTM in terms of accuracy and stability. Additionally, network parameter count poses a significant challenge for hardware implementation; the proposed method addresses this by achieving substantial parameter reduction alongside enhanced adaptability and improved precision under varying environmental conditions. This approach could serve as an effective solution for facilitating future on-orbit servicing and space debris management missions.

Keywords:*Orbital rendezvous, Meta-reinforcement learning, Transformer, On-orbit servicing.*

۱. مقدمه

هدف اصلی در ملاقات مداری، رساندن دو فضاپیما که به دور یک جرم سماوی مرکزی در حال گردش هستند، به یک پیکربندی نسبی از پیش تعیین‌شده در مجاورت یکدیگر است. در این فرآیند، یکی از فضاپیماها، موسوم به "دنبال‌کننده"، از طریق انجام مانورهای فعال به مدار فضاپیمای دیگر، معروف به "هدف"، منتقل می‌شود و سپس به موقعیت نسبی مشخصی در نزدیکی هدف با سرعتی ایمن نزدیک می‌شود. هدایت این عملیات مستلزم بهینه‌سازی مصرف سوخت، زمان ملاقات و کنترل دقیق است که معمولاً در قالب یک مسئله بهینه‌سازی کنترل (OCP) با استفاده از روش‌هایی مانند برنامه‌ریزی خطی، برنامه‌ریزی درجه دوم و سایر روش‌های مشابه مدل‌سازی می‌شود[۱, ۲]. هدایت حلقه‌باز در محیط فضایی به دلیل حساسیت بالا نسبت به عدم‌قطعیت‌های محیطی، چالش‌برانگیز است. بنابراین، کنترل خودکار معمولاً به یک سیستم بازخورد حلقه‌بسته نیاز دارد تا بتواند از تخمین بلادرنگ حالت توسط سیستم ناوبری به عنوان ورودی استفاده کند. در این حالت، مسئله بهینه‌سازی باید به‌طور پیوسته با استفاده از فیلترهای ناوبری به‌روزرسانی شود، که این امر مستلزم کارایی محاسباتی بالا است. کنترل پیش‌بینی مدل (MPC) به عنوان یک روش کنترل بازخوردی حلقه‌بسته، بهبود دقت و خودمختاری را فراهم می‌کند [3] و می‌تواند با استفاده از چندین حسگر، طراحی هدایت اولیه را از سیستم ناوبری مستقل سازد، مگر اینکه عدم‌قطعیت در مشاهده به‌عنوان نویز افزوده مدل شود [۴].

در سال‌های اخیر، روش‌های بهینه‌سازی محدب مانند برنامه‌ریزی مخروطی و حل‌کننده‌های نقطه داخلی برای بهینه‌سازی انتقال مداری، به‌ویژه در کاربردهایی مانند فرودهای مرحله اول و فرودهای سیارهای، مورد استفاده قرار گرفته‌اند[۵, ۶]. روش‌های نقطه داخلی، بدون نیاز به مقادیر اولیه خاص، همگرایی بهینه را در زمان چندجمله‌ای تضمین می‌کنند. اگرچه این روش‌ها در چارچوب MPC در سیستم‌های داخل‌پروازی دارای قابلیت‌های بالقوه هستند [۷, ۸]، اما تبدیل مسائل غیرخطی به مسائل محدب متوالی ممکن است حل مسئله را کند کرده و فرکانس به‌روزرسانی‌های حلقه‌بسته را محدود کند.

یادگیری تقویتی عمیق (RL) به عنوان یک چارچوب پیشرفته در هوش مصنوعی، به ویژه در حوزه کاربردهای فضایی، جایگزین مناسبی برای روش‌های سنتی مانند کنترل پیش‌بین مدل‌محور (MPC) شده است. این روش در زمینه‌های متنوعی از جمله انتقالات بین سیارهای [۹, ۱۰]، مأموریت‌های میانماه-زمین [۱۱,۱۳]، فرودهای سیاره‌ای [۱۴, ۱۵] و مانورهای ملاقات و رهگیری نهایی مورد استفاده قرار گرفته است. یادگیری تقویتی از طریق شبکه‌های عصبی عمیق (DNN) که برای نگاشت مشاهدات به اقدامات کنترلی آموزش داده می‌شوند، هدایت و کنترل فضاپیما را خودکار کرده و امکان بهینه‌سازی عملکرد را در محیط‌های شبیه‌سازی‌شده فراهم می‌کند. الگوریتم‌هایی مانند بهینه‌سازی سیاست مجاورتی (PPO) با افزایش پایداری در به‌روزرسانی سیاست‌ها، برای حل مسائل کنترل پیوسته در RL بسیار مؤثر عمل می‌کنند [۱۶, ۱۷].

یادگیری تقویتی با بهره‌گیری از الگوریتم‌های اکتشافی که در طی شبیه‌سازی‌های متعدد آموزش می‌بینند، مقاومت سیستم را در برابر تغییرات محیطی افزایش داده و عملکرد کنترل را بهبود می‌بخشد. در مأموریت‌های چندهدفه، هر مسیر به عنوان نمونه‌ای از توزیع انتقال‌های پیوسته با زمان پرواز قابل قبول میان مدارهای جفت هدف در نظر گرفته می‌شود. این رویکرد نشان می‌دهد که الگوریتم‌های یادگیری تقویتی متا[1] با تسریع

سازگاری با وظایف جدید و استفاده از دانش پیشین، عملکرد بهینه‌ای دارند. ترکیب این روش‌ها با شبکه‌های عصبی بازگشتی (RNN) مانند GRU و LSTM، امکان یادگیری روابط زمانی را فراهم کرده و در مقایسه با شبکه‌های پرسپترون چندلایه (MLP)، تطابق بهتری در محیط‌های نامطمئن و جزئی‌مشاهده‌پذیر مانند فرود خودکار در ماه و عملیات در نزدیکی سیارک‌ها نشان می‌دهند. در طراحی مانورهای خودمختار پهلوگیری فضاپیما در شش درجه آزادی (شامل حرکت خطی و چرخشی در سه محور)، یادگیری تقویتی به عنوان یک چارچوب هوش مصنوعی، دقت و کارایی عملیات را از طریق بهینه‌سازی تصمیم‌گیری در محیط‌های پویا و پیچیده افزایش می‌دهد. این روش امکان تطبیق‌پذیری با شرایط غیرقطعی را فراهم کرده و عملکرد سیستم را در مواجهه با تغییرات محیطی بهبود می‌بخشد[۱۸]. از سوی دیگر، متا-یادگیری به عنوان یک رویکرد قدرتمند، این قابلیت را به سیستم می‌دهد که از تجربیات گذشته در شرایط مختلف مأموریت یاد گرفته و این دانش را به شرایط جدید تعمیم دهد. این روش در بهبود هدایت تطبیقی فضاپیما در مأموریت‌های ملاقات مداری با پیشرانه محدود به کار گرفته شده و نشان داده است که چگونه می‌توان با بهره‌گیری از توانایی تعمیم‌پذیری متا-یادگیری، عملکرد سیستم را در محیط‌های پویا و غیرقطعی بهینه کرد [۱۹].

ترکیب یادگیری تقویتی و متا-یادگیری، به ویژه در مأموریت‌های فضایی پیچیده، توانایی سیستم‌های خودمختار را برای انطباق با شرایط متغیر و بهبود عملکرد کلی مأموریت‌ها به‌طور چشمگیری افزایش می‌دهد. این رویکردها نه تنها کارایی سوخت و دقت عملیات را بهبود می‌بخشند، بلکه مقاومت سیستم را در برابر عدم قطعیت‌های محیطی نیز تقویت می‌کنند.

ترنسفورمرها نیز در چند سال اخیر به دلیل توانایی در پردازش داده‌های ترتیبی و مقیاس‌پذیری در داده‌های حجیم، به عنوان عاملی مؤثر در RL ظاهر شده‌اند. وزن‌های توجه به خود در ترنسفورمرها به عنوان پارامترهای وابسته به زمینه عمل کرده و به عامل RL کمک می‌کنند تا در محیط‌های مختلف با استفاده از اطلاعات ترتیبی گذشته به وظایف جدید تطبیق یابد و به این ترتیب، یادگیری تقویتی متا را تقویت کنند[۲۰, ۲۱]. این معماری‌ها در ترکیب اطلاعات فضایی و زمانی برای هدایت و کنترل فضاپیما در عملیات های پیچیده GNC بسیار کارآمدتر از شبکه‌های بازگشتی عمل کرده‌اند. این مقاله به بررسی دو روش یکی با استفاده از شبکه LSTM و دیگری با استفاده از یک شبکه ترنسفورمر در یادگیری تقویتی متا برای حل مسائل هدایت و کنترل فضاپیما پرداخته و عملکرد آن ها را در محیط‌های پویا با اجراهای مونت کارلو به عنوان یک مدل مقاوم برای هدایت خودمختار تحلیل می‌کند.

۲. روابط حاکم و روش حل مسئله

در مسئله‌ی ملاقات مداری، هدف اصلی بهینه‌سازی مصرف سوخت و اطمینان از رسیدن کارآمد فضاپیما به هدف است. این مسئله را می‌توان از طریق فرمول‌بندی یک مسئله‌ی کنترل بهینه مورد بررسی قرار داد [۲۲]و[۲۳]. در این راستا، تابع هزینه به صورت کمینه‌سازی مجموع مربعات نیروهای کنترلی و گشتاورهای اعمال‌شده در طول زمان تعریف می‌شود. این تابع هزینه مطابق معادله (۱) به شکل زیر بیان می‌گردد:

$$min \int_{t_0}^{t_f} \frac{1}{2}\Big((F_x^H)^2 + \big(F_y^H\big)^2 + (F_z^H)^2 + (T_x^C)^2 + \big(T_y^C\big)^2 + (T_z^C)^2\Big)\, dt \qquad (۱)$$

این فرمول نشان‌دهنده‌ی هدف کمینه‌سازی مجموع مربعات نیروهای کنترلی و گشتاورها در طول زمان است. در حقیقت، این تابع هزینه بیانگر این است که تلاش‌های کنترلی در سه محور

مختلف (محورهای x، y و z) تا حد ممکن کوچک شوند. این هدف با کمینه‌سازی انتگرال مجموع مربعات نیروهای خطی (F_x^H، F_y^H و F_z^H) و گشتاورهای چرخشی (T_x^C، T_y^C و T_z^C) از زمان شروع (t_0) تا زمان پایانی (t_f) حاصل می‌شود. در اینجا، بردار نیروی کنترلی در چارچوب هیل با ($\vec{F}^H$) نمایش داده شده و بردار گشتاور در چارچوب بدنه فضاپیمای تعقیب‌کننده با ($\vec{T}^C$) نشان داده می‌شود. هدف از کمینه‌سازی تلاش کنترلی، حفظ سوخت و اطمینان از رسیدن کارآمد فضاپیما به هدف است.

۲-۱. دینامیک مداری نسبی

حرکت نسبی فضاپیمای تعقیب‌کننده نسبت به فضاپیمای هدف را می‌توان با استفاده از معادلات کلُهِسی–ویلتیِر–هیل (CWH)[۲] مدل‌سازی کرد. این معادلات به صورت زیر بیان می‌شوند:

$$\ddot{x} = 2\Omega\dot{y} + 3\Omega^2 x + \frac{F_x^H}{m_C} \quad (۲)$$

$$\ddot{y} = -2\Omega\dot{x} + \frac{F_y^H}{m_C} \quad (۳)$$

$$\ddot{z} = -\Omega^2 z + \frac{F_z^H}{m_C} \quad (۴)$$

در این معادلات، $(r = [x, \ y, \ z]^T)$ بردار موقعیت نسبی فضاپیمای تعقیب‌کننده نسبت به فضاپیمای هدف، Ω سرعت زاویه‌ای چارچوب هیل نسبت به چارچوب لخت، و m_C جرم فضاپیمای تعقیب‌کننده است. این معادلات دینامیک انتقالی نسبی را در یک مدار دایره‌ای توصیف می‌کنند و برای طراحی کنترل‌کننده‌های مداری کاربرد گسترده‌ای دارند.

۲-۲. دینامیک چرخشی

دینامیک چرخشی فضاپیمای تعقیب‌کننده و فضاپیمای هدف با استفاده از معادلات (۵) تا (۱۰) مدل‌سازی می‌شود. برای فضاپیمای تعقیب‌کننده[۳]، معادلات دینامیک چرخشی به صورت زیر هستند:

$$\dot{\omega}_x^C = \frac{(I_{yy}^C - I_{zz}^C)\omega_z^C\omega_y^C + T_x^C}{I_{xx}^C} \quad (۵)$$

$$\dot{\omega}_y^C = \frac{(I_{zz}^C - I_{xx}^C)\omega_z^C\omega_x^C + T_y^C}{I_{yy}^C} \quad (۶)$$

$$\dot{\omega}_z^C = \frac{(I_{xx}^C - I_{yy}^C)\omega_y^C\omega_x^C + T_z^C}{I_{zz}^C} \quad (۷)$$

برای فضاپیمای هدف[۴]، معادلات مشابهی بدون حضور گشتاورهای کنترلی اعمال می‌شود:

$$\dot{\omega}_x^T = \frac{(I_{yy}^T - I_{zz}^T)\omega_z^T\omega_y^T}{I_{xx}^T} \quad (۸)$$

$$\dot{\omega}_y^T = \frac{(I_{zz}^T - I_{xx}^T)\omega_z^T\omega_x^T}{I_{yy}^T} \quad (۹)$$

$$\dot{\omega}_z^T = \frac{(I_{xx}^T - I_{yy}^T)\omega_y^T\omega_x^T}{I_{zz}^T} \quad (۱۰)$$

در این معادلات، $\vec{\omega}^i$ مؤلفه‌های سرعت زاویه‌ای فضاپیما i (که می‌تواند Chaser یا Target باشد) و I^i مؤلفه‌های اصلی ماتریس اینرسی فضاپیما هستند .سرعت‌های زاویه‌ای فضاپیمای تعقیب‌کننده (C) و فضاپیمای هدف (T) می‌توانند نسبت به چارچوب مختصات هیل به صورت $H_{\vec{\omega}^i}$ در محورهای اصلی بیان شوند که با استفاده از یک تبدیل مشخص به صورت معادله (۱۱) محاسبه می‌شود.

$$H_{\vec{\omega}^i} = \vec{\omega}^i - R_H^i \begin{bmatrix} 0 \\ 0 \\ \Omega \end{bmatrix} \quad (۱۱)$$

که در آن ماتریس تبدیل R_H^i طبق معادله‌ی (۱۲) تعریف شده است:

$$R_H^i = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \quad (۱۲)$$

در اینجا پارامترهای R_{ij} به صورت زیر تعریف می‌شوند:

$$\begin{aligned} R_{11} &= (q_4^i)^2 + (q_1^i)^2 - (q_2^i)^2 - (q_3^i)^2 \\ R_{12} &= 2(q_1^i q_2^i + q_3^i q_4^i) \\ R_{13} &= 2(q_1^i q_3^i - q_2^i q_4^i) \\ R_{21} &= 2(q_1^i q_2^i + q_3^i q_4^i) \\ R_{22} &= (q_4^i)^2 - (q_1^i)^2 + (q_2^i)^2 - (q_3^i)^2 \\ R_{23} &= 2(q_2^i q_3^i + q_1^i q_4^i) \\ R_{31} &= 2(q_1^i q_3^i + q_2^i q_4^i) \\ R_{32} &= 2(q_2^i q_3^i + q_1^i q_4^i) \\ R_{33} &= (q_4^i)^2 - (q_1^i)^2 - (q_2^i)^2 + (q_3^i)^2 \end{aligned} \quad (۱۳)$$

که در آن $\vec{q}^{i}$ بردار کواترنیون فضاپیمای i نسبت به چارچوب هیل است. با استفاده از تعریف ماتریس تبدیل (۱۲)، معادله‌ی (۱۱) به صورت زیر بازنویسی می‌شود:

$$H_{\vec{\omega}^i} = \vec{\omega}^i - \Omega \begin{bmatrix} 2(q_1^i q_3^i - q_2^i q_4^i) \\ 2(q_2^i q_3^i + q_1^i q_4^i) \\ (q_4^i)^2 - (q_1^i)^2 - (q_2^i)^2 + (q_3^i)^2 \end{bmatrix} \quad (۱۴)$$

جهت‌گیری فضاپیما با استفاده از محورهای اصلی نسبت به چارچوب مختصات هیل قابل نمایش است و این جهت‌گیری با یک کواترنیون توصیف می‌شود. معادله دیفرانسیل سینماتیکی کواترنیون به صورت زیر است:

$$\dot{\vec{q}}^i = \frac{1}{2} \begin{bmatrix} 0 & H_{\omega_z^i} & -H_{\omega_y^i} & H_{\omega_x^i} \\ -H_{\omega_z^i} & 0 & H_{\omega_x^i} & H_{\omega_y^i} \\ H_{\omega_y^i} & -H_{\omega_x^i} & 0 & H_{\omega_z^i} \\ -H_{\omega_x^i} & -H_{\omega_y^i} & -H_{\omega_z^i} & 0 \end{bmatrix} \vec{q}^i \quad (۱۵)$$

برای جلوگیری از برخورد فضاپیماها، شرط ایمنی (۱۶) تعریف می‌شود که اطمینان حاصل کند فاصله بین فضاپیماها همواره از یک مقدار حداقل ایمن بیشتر باشد. این شرط به شکل زیر است:

$$r^2 - (x^2 + y^2 + z^2) < 0 \quad (۱۶)$$

که در آن r^2 مقدار مرزی برای فاصله‌ایمن بین فضاپیماها است. این شرط بیان می‌کند که مجذور فاصله نسبی بین دو فضاپیما باید همواره از r^2 کمتر باشد، به عبارت دیگر، فاصله واقعی بین دو فضاپیما باید از r (ریشه دومr^2) بیشتر باشد. در این مسئله، مقدار $r^2 = 4m^2$ فرض شده است، که به معنای فاصله‌ی ایمن حداقل $r = 2m$ بین دو فضاپیما است. حفظ فاصله‌ی ایمن بین فضاپیماها در مأموریت‌های فضایی از اهمیت بالایی برخوردار است، زیرا برخورد فضاپیماها می‌تواند منجر به آسیب‌های جبران‌ناپذیر به تجهیزات، از دست رفتن مأموریت، و حتی خطر برای خدمه (در مأموریت‌های سرنشین‌دار) شود. علاوه بر این، فاصله‌ی ایمن به سیستم‌های کنترل و ناوبری اجازه می‌دهد تا با دقت بیشتری عملیات ملاقات و پهلوگیری را انجام دهند. همچنین، در مدل‌سازی دینامیک فضاپیماها همواره خطاهایی وجود دارد، و فاصله‌ی ایمن به عنوان یک حاشیه‌ی اطمینان عمل می‌کند که اثرات این خطاها را کاهش می‌دهد. بنابراین، اعمال این شرط در طراحی کنترل‌کننده‌ها و الگوریتم‌های ناوبری، نقش کلیدی در موفقیت مأموریت‌های فضایی ایفا می‌کند.

۳. یادگیری تقویتی عمیق

در این بخش ابتدا به بیان تعریف فرآیند تصمیم گیری مارکوف و سپس الگوریتم PPO و شبکه های عصبی و LSTM می پرازیم و در نهایت به سراغ ترنسفورمرها می رویم.

۳-۱. فرایند تصمیم‌گیری مارکوف با مشاهده ناقص (POMDP)

POMDP به عنوان یک تعمیم از فرایند تصمیم‌گیری مارکوف (MDP) در نظر گرفته می‌شود. تفاوت اصلی این دو در این است که در POMDP، عامل تصمیم‌گیرنده تنها به بخشی از اطلاعات مربوط به وضعیت محیط دسترسی دارد و به همین دلیل، مشاهده‌های او ناقص هستند [۲۴]. در چنین شرایطی، عامل تنها می‌تواند از طریق مشاهدات جزئی، اطلاعاتی محدود درباره محیط به دست آورد. مسئله POMDP به‌صورت یک شش‌تایی $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \Omega, \mathcal{O})$ توصیف می‌شود، که در آن $\mathcal{S}$ فضای تمامی حالات ممکن در محیط، $\mathcal{A}$ فضای تمامی اقدامات ممکن محیط، $\mathcal{R}_{\mathcal{A}}(\mathrm{s}, \acute{\mathrm{s}})$ تابع پاداش است که مقدار پاداش را برای انتقال از حالت s به حالت ś مشخص می‌کند ، $\mathcal{T}_{\mathcal{A}}(\mathrm{s}, \acute{\mathrm{s}})$ تابع انتقال حالت است که احتمال انتقال از حالت s به حالت ś با انجام عمل a ، Ω فضای تمامی مشاهدات ممکن و $\mathcal{O}(\acute{\mathrm{o}}|\mathrm{a}, \acute{\mathrm{s}})$ تابع مشاهده است که احتمال مشاهده ó را پس از انجام عمل a و انتقال به حالت ś، تعریف می‌گردند. برای

حل دقیق مسئلهPOMDP، لازم است اقدام بهینه برای هر مشاهده ممکن در تمام حالات ممکن انتخاب شود. این اقدام بهینه، پاداش مورد انتظار عامل را در یک افق زمانی (که ممکن است نامحدود باشد) به حداکثر می‌رساند. با این حال، حل دقیق مسئله POMDP تنها برای دسته‌ای محدود از مسائل امکان‌پذیر است، زیرا این مسئله از نظر محاسباتی بسیار پیچیده است. به همین دلیل، معمولاً از روش‌های تقریبی برای ساده‌سازی و حل مسئله استفاده می‌شود. یکی از روش‌های متداول برای ساده‌سازی مسئله POMDP، تبدیل آن به یک مسئله MDP در فضای باور[5] است. در این فرمول‌بندی جدید، مسئله به‌صورت یک چهارتایی $(\mathcal{B}, \mathcal{A}, \mathcal{R}_{\mathcal{A}}, \mathcal{T}_{\mathcal{A}})$ تعریف می‌شود. در این فرمول‌بندی، $\mathcal{B}$ فضای باور است و باور $b = p(s|h)$ به عنوان احتمال حضور در حالت s پس از تاریخچه h از حالات قبلی تعریف می‌شود. $\mathcal{A}$ همان فضای اقدامات اصلی است. $\mathcal{R}_{\mathcal{A}}(b, \acute{b})$ تابع پاداش در فضای باور است که پاداش انتقال از باور b به باور $\acute{b}$ را مشخص می‌کند و $\mathcal{T}_{\mathcal{A}}(b, \acute{b})$ تابع انتقال باور است که احتمال رسیدن به باور جدید $\acute{b}$ از باور b با انجام عمل a را نشان می‌دهد. هدف در این فرمول‌بندی، همچنان حداکثرسازی پاداش بلندمدت است. سیاست بهینه در مسئله POMDP به دو شکل تعریف می‌شود: سیاست بهینه برای مسائل با افق زمانی محدود و سیاست بهینه برای مسائل با افق زمانی نامحدود. در این پژوهش، مسئله مورد بررسی در دسته مسائل با افق نامحدود قرار می‌گیرد و سیاست بهینه آن با استفاده از ضریب تخفیف $\gamma \in [0,1]$ مطابق معادله (۱۷) تعریف می‌شود

$$\pi_* = \underset{\pi}{\operatorname{argmax}}\, \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k \mathcal{R}_{\mathcal{A}}(b_k, b_{k+1})\right] \qquad (۱۷)$$

۳-۲. بهینه‌سازی سیاست پروگزیمال (PPO)[6]

در این پژوهش، از فرمول‌بندی الگوریتم بهینه‌سازی سیاست مجاور (PPO) که توسط شولمن و همکاران در سال ۲۰۱۷ ارائه شده است [۱۶]، استفاده شده است. الگوریتم PPO یک روش پیشرفته در حوزه یادگیری تقویتی عمیق (DRL) محسوب می‌شود و به‌عنوان یکی از کارآمدترین الگوریتم‌های گرادیان سیاست در بسیاری از مسائل مرجع شناخته می‌شود. این الگوریتم از ترکیب روش‌های A2C (Advantage Actor-Critic) و بهینه‌سازی سیاست ناحیه اعتماد (TRPO) که در [۲۵] توسعه یافته است، مشتق شده است. PPO مسئله بهینه‌سازی را به گونه‌ای فرمول‌بندی می‌کند که گام‌های گرادیان با اعمال محدودیت‌هایی تنظیم شوند تا از تغییرات ناگهانی و ناپایدار در سیاست جلوگیری شود.

یکی از ویژگی‌های کلیدی PPO، استفاده از یک تابع هدف بریده شده[7] در فرآیند بهینه‌سازی سیاست است. این تابع هدف با استفاده از نسبت احتمالی $p_k(\theta)$ که در معادله (۱۸) تعریف شده است، محاسبه می‌شود:

$$p_k(\theta) = \frac{\pi_{\theta,k}(u_k|x_k)}{\pi_{\theta,k-1}(u_k|x_k)} \qquad (۱۸)$$

نسبت احتمالی $p_k(\theta)$ تغییرات در سیاست را نسبت به نسخه قبلی آن اندازه‌گیری می‌کند. با محدود کردن این نسبت در بازه $[1-\epsilon, 1+\epsilon]$ ، الگوریتم از تغییرات بیش از حد در سیاست جلوگیری می‌کند و به همگرایی پایدار کمک می‌نماید. تابع هدف بریده‌شده در PPO به صورت زیر تعریف می‌شود:

$$L_{CLIP}(\theta) = \mathbb{E}_{p(\tau)}[min[p_k(\theta), clip(p_k(\theta), 1-\epsilon, 1+\epsilon)]A_w^{\pi}(u_k, x_k)] \qquad (۱۹)$$

در این معادله، $A_w^{\pi}(u_k, x_k)$ تابع مزیت[8] است که نشان می‌دهد یک عمل خاص چقدر بهتر از

میانگین اعمال ممکن در حالت x_k است. این تابع به صورت زیر تعریف می‌شود:

$$A_w^{\pi}(u_k, x_k) = \left[\sum_{j=k}^{T} \gamma^{j-k} r(x_j, u_j)\right] - V_w^{\pi}(x_k) \quad (۲۰)$$

در اینجا، $\gamma \in [0,1]$ ضریب تخفیف و $V_w^{\pi}(x_k)$ تابع ارزش[۹] است که توسط شبکه منتقد محاسبه می‌شود. تابع ارزش نشان‌دهنده ارزش مورد انتظار حالت x_k تحت سیاست π است.

به‌علاوه، برای افزایش سطح کاوش در طول فرآیند آموزش، یک ترم منظم‌کننده آنتروپی[۱۰] به تابع هدف اضافه می‌شود [۲۶,۱۶]. بنابراین، تابع هدف نهایی به صورت زیر تعریف می‌گردد:

$$L_p = L_{CLIP}(\theta) + c_2 S[\pi_\theta](x_k) \quad (۲۱)$$

در این معادله، $S[\pi_\theta](x_k)$ آنتروپی سیاست و c_2 ضریب تنظیم‌کننده آنتروپی است.

تابع ارزش $V_w^{\pi}(x_k)$ نیز با استفاده از یک تابع هزینه بهینه‌سازی می‌شود که به صورت زیر تعریف شده است:

$$L_w = \sum_{i=1}^{N} \left(V_w^{\pi}(x_k^i) - \left[\sum_{j=k}^{T} \gamma^{j-k} r(x_j^i, u_j^i)\right] \right)^2 \quad (۲۲)$$

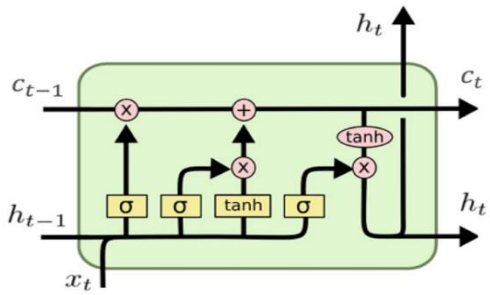
در اینجا، N تعداد مسیرهای استفاده‌شده برای به‌روزرسانی شبکه را نشان می‌دهد. هر مسیر شامل یک سری از مشاهدات، اعمال و پاداش‌ها است که از تعامل عامل با محیط جمع‌آوری شده‌اند.

الگوریتم PPO به دلیل کارایی بالا در مسائل با فضای حالت و عمل پیوسته، به‌طور گسترده در برنامه‌ریزی مسیرهای فضایی و سایر کاربردهای پیچیده یادگیری تقویتی مورد استفاده قرار می‌گیرد [۲۷]. در این پژوهش، از شبکه‌های عصبی مصنوعی به‌عنوان تقریب‌زننده توابع سیاست و ارزش استفاده شده است. جزئیات مربوط به معماری شبکه‌های عصبی در بخش‌های بعدی ارائه خواهد شد.

در این پژوهش، دو معماری اصلی شبکه عصبی برای تعریف عامل هدایت و کنترل خودکار طراحی شده‌اند: اولی: "شبکه عصبی بازگشتی (RNN)" و دومی "شبکه عصبی ترنسفورمر (Transformer)".

۳-۳. شبکه عصبی بازگشتی (RNN)

شبکه‌های عصبی بازگشتی (RNN) برای پردازش داده‌های متوالی و وابسته به زمان طراحی شده‌اند. برخلاف شبکه‌های عصبی تغذیه پیشرو (FFNN)[۱۱]، RNNها معرفی‌شده در [۲۸] دارای حلقه‌های بازخوردی هستند که به آن‌ها امکان می‌دهد اطلاعات مربوط به مراحل قبلی را در حافظه خود نگه دارند. در میان انواع RNN، "لایه‌های بازگشتی حافظه کوتاه‌مدت-بلندمدت (LSTM) " به‌کار گرفته شده‌اند. ساختار یک سلول LSTM در شکل ۱ نشان داده شده است.



شکل ۱- یک سلولLSTM

RNNها به‌دلیل توانایی خود در ذخیره اطلاعات مربوط به حالت‌های گذشته، برای برنامه‌ریزی مسیرهای امن و کنترل خودکار بسیار مناسب هستند. این ویژگی به عامل اجازه می‌دهد تا سریع‌تر به اهداف مأموریت دست یابد و از برخورد با موانع جلوگیری کند. همان‌طور که در [۲۹] توضیح داده شده است، ایده پشت فرمول‌بندی PPO با استفاده از شبکه عصبی بازگشتی در مقایسه با مدل ساده‌تر تغذیه پیشرو به مزایای بالقوه‌ای مرتبط است که چنین معماری می‌تواند از نظر پایداری آموزش و عملکرد مناسب تر باشد. در واقع، با داشتن قابلیت ذخیره اطلاعات

حالت‌های گذشته، این معماری می‌تواند به شدت بر برنامه‌ریزی مسیرهای امن عامل تأثیر بگذارد تا سریع‌تر به اهداف مأموریت دست یابد.

۳-۴. ترنسفورمر

شبکه‌های عصبی ترنسفورمر بر اساس مکانیسم "توجه به خود [۳۵][۱۲] " استوار هستند. این معماری در سال ۲۰۱۷ توسط واسوانی و همکاران [۳۴] معرفی شد و در حوزه‌هایی مانند پردازش زبان طبیعی (NLP)[۳۰, ۳۱] و تشخیص تصویر[۳۲, ۳۳] موفقیت‌های چشمگیری داشته‌است.

مکانیسم توجه به خود به مدل اجازه می‌دهد تا وابستگی‌ها بین عناصر مختلف یک توالی ورودی را استخراج کند. این ویژگی باعث می‌شود که مدل بتواند روی اطلاعات مرتبط تمرکز کند و فرآیند استخراج ویژگی‌ها را تسهیل نماید. برخلاف شبکه‌های عصبی بازگشتی (RNN) مانند LSTM [۳۶]، ترنسفورمرها نیازی به فشرده‌سازی اطلاعات در حالت‌های پنهان ندارند و از مشکلاتی مانند محو یا انفجار گرادیان جلوگیری می‌کنند. این ویژگی‌ها به ترنسفورمرها امکان می‌دهند تا وابستگی‌های بلندمدت را به‌طور مؤثرتری مدل‌سازی کنند. همچنین، پردازش غیرتوالی داده‌ها در ترنسفورمرها، محاسبات موازی و فاز آموزش سریع‌تری را فراهم می‌آورد.

یک شبکه ترنسفورمر از چندین لایه یکسان تشکیل شده است. هر لایه به‌صورت یک تابع غیرخطی دنباله به دنباله عمل می‌کند و یک دنباله ورودی متشکل از n بردار با d-بعد $I = [i_1^T, i_2^T, \dots, i_n^T] \in \mathbb{R}^{n\times d}$ را به یک دنباله خروجی $H = [h_1^T, h_2^T, \dots, h] \in \mathbb{R}^{n\times d}$ تبدیل می‌کند که شامل بردارهای واقعی با بعد d است. هسته هر مدل ترنسفورمر، بلوک‌های توجه هستند. تابع توجه، یک "کوئری" را به مجموعه‌ای از جفت‌های "کلید-مقدار" نگاشت می‌دهد. خروجی به‌صورت یک میانگین وزنی از مقادیر محاسبه می‌شود:

$$\alpha(\beta(Q,K),V,\chi) = \sum\left(\frac{\chi(\beta(Q,K))}{\sqrt{d_k}}\right)V \qquad (۲۳)$$

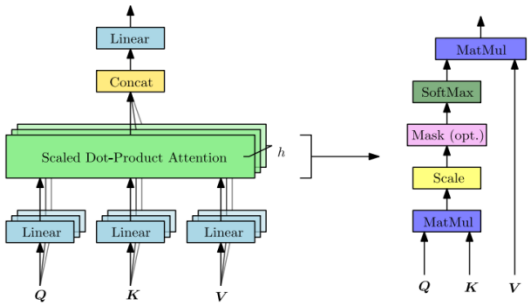
که در آن Q ماتریس کوئری‌ها، K ماتریس کلیدها، V ماتریس مقادیر، χ تابع ماسک، β تابع سازگاری (ضرب داخلی QK^T) است. در ترنسفورمرها، کوئری‌ها، کلیدها و مقادیر به چندین زیرفضای کوچکتر تقسیم می‌شوند و تابع توجه به‌طور موازی روی هر یک از این زیرفضاها اعمال می‌شود. خروجی بلوک توجه چندسر به صورت زیر محاسبه می‌شود:

$$\xi(Q,K,V,\chi,h) = [\alpha_1,\cdots,\alpha_h]W_o \qquad (۲۴)$$

که در آن هر سر توجه α_i به صورت زیر تعریف می‌شود:

$$\alpha_i = \alpha(\beta(QW_{Q,i},KW_{k,i}),VW_{v,i},\chi) \qquad (۲۵)$$

که در آن پروجکشن‌ها، ماتریس‌های وزنی یادگرفته‌شده $W_{Q,i}, W_{k,i}, W_{v,i} \in \mathbb{R}^{d\times d/h}$ و $W_o \in \mathbb{R}^{d\times d}$ هستند.



شکل ۲- توجه چندسر[۱۳] (راست) و توجه مقیاس شده بر پایه ضرب داخلی (چپ)

شکل ۲ ساختار توجه چندسر و توجه مقیاس‌شده بر پایه ضرب داخلی را نشان می‌دهد. در بخش چپ، مراحل محاسبه توجه مقیاس‌شده شامل ضرب ماتریسی کوئری (Q) و کلید (K)، اعمال تابع SoftMax و محاسبه خروجی با استفاده از مقادیر (V) نمایش داده شده است. در بخش راست، توجه چندسر با تقسیم کوئری، کلید و

مقادیر به چندین سر و انجام محاسبات موازی به تصویر کشیده شده است.

۴. سناریو ماموریت

در این سناریو، مرحله نهایی نزدیک‌شدن یک فضاپیمای دنبال‌کننده به یک هدف فضایی مورد بررسی قرار می‌گیرد. هدف اصلی این سناریو، هدایت فضاپیمای دنبال‌کننده به سمت هدف بدون وقوع برخورد است. فضاپیمای دنبال‌کننده به‌عنوان یک جسم مکعبی شکل با چگالی یکنواخت، طول ۱ متر و جرم ۱۰۰ کیلوگرم مدل‌سازی شده است. این فضاپیما مجهز به یک سامانه کنترل موقعیت و مدار پیشرفته است که شامل رانشگرها و چرخ‌های واکنشی می‌شود. رانشگرها قادر به تولید نیروی خالصی معادل ± 10 نیوتن در امتداد هر یک از سه محور اصلی بدنه هستند، در حالی که چرخ‌های واکنشی می‌توانند گشتاور خالصی معادل ± 0.2 نیوتن‌متر را در امتداد هر سه محور اعمال کنند. این محدودیت‌ها بر اساس قابلیت‌های سامانه‌های تجاری موجود برای فضاپیماهای کوچک تعیین شده‌اند [۳۷].

فضاپیمای دنبال‌کننده با بهره‌گیری از مجموعه‌ای از محرک‌ها، کنترل کامل موقعیت، سرعت، وضعیت و نرخ چرخش خود را در اختیار دارد. همچنین، این فضاپیما به ابزارهای ناوبری دقیقی مجهز است که امکان اندازه‌گیری وضعیت نسبی آن نسبت به هدف را فراهم می‌کند. در طول فرآیند نزدیک‌شدن، ابزارهای ناوبری باید به‌طور مداوم به سمت هدف نشانه‌گیری شوند. هدف این سناریو به‌صورت یک مکعب یکنواخت با دو بازوی بلند (شبیه‌ساز آرایه‌های خورشیدی) مدل‌سازی شده است. برای جلوگیری از برخورد، یک ناحیه ایمنی کروی با شعاع ۴ متر در اطراف هدف تعریف شده است که شامل یک راهرو ورودی مخروطی برای فضاپیمای دنبال‌کننده می‌باشد. پس از ورود به این راهرو، فضاپیما باید مسیر خود را ادامه داده و به موقعیت مشخصی در نزدیکی هدف برسد، به‌گونه‌ای که شرایط نهایی از جمله کاهش موقعیت، سرعت و وضعیت نسبت به هدف را برآورده کند. این شرایط نهایی، که در جدول ۱ ارائه شده‌اند، برای آغاز فاز الحاق ضروری هستند. موفقیت این مأموریت به دستیابی به شرایط نهایی بدون وقوع هرگونه برخورد با ناحیه ایمنی بستگی دارد.

جدول ۱- شرایط پایانی سناریو

شرایط	پارامتر
$r_e < 0.5\ m$	موقعیت
$v_e < 0.1\ \ m/s$	سرعت
$\theta_e < 5\ deg$	وضعیت

۴-۱. دینامیک مسئله

دینامیک مسئله چگونگی تغییر وضعیت s محیط با گذر زمان را مشخص می‌کند. در این سناریو، وضعیت محیط توسط موقعیت، سرعت و وضعیت (جهت‌گیری) فضاپیمای دنبال‌کننده و هدف تعریف می‌شود. موقعیت و سرعت در چارچوب مرجع محلی افقی (LVLH) بیان شده‌اند که برای سناریوهای ملاقات فضایی مناسب است. در این چارچوب مرجع، هدف در مبدأ قرار دارد، بنابراین موقعیت (r) و سرعت (v) دنبال‌کننده نسبت به هدف به صورت زیر تعریف می‌شوند:

$$r = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}_{LVLH}, v = \begin{bmatrix} \dot{r}_x \\ \dot{r}_y \\ \dot{r}_z \end{bmatrix}_{LVLH} \quad (۲۶)$$

جهت‌گیری دو فضاپیما با استفاده از کواترنیون‌های واحد بیان می‌شود. کواترنیون واحد یک آرایه چهار عضوی است که می‌تواند هر چرخشی را در فضای سه‌بعدی نمایش دهد و از این رو برای نمایش جهت‌گیری یک چارچوب مرجع نسبت به چارچوب مرجع دیگر استفاده می‌شود. در این سناریو، کواترنیون q_c جهت‌گیری

چارچوب بدنه دنبال‌کننده و q_t جهت‌گیری چارچوب بدنه هدف را نسبت به چارچوب LVLH نشان می‌دهند. همچنین، نرخ‌های چرخش چارچوب‌های بدنه دنبال‌کننده و هدف نسبت به چارچوب LVLH به ترتیب با دو بردار ω_c و ω_t بیان می‌شوند. بردار وضعیت کامل سیستم S به صورت زیر تعریف می‌شود:

$$S = \begin{bmatrix} r \\ v \\ q_c \\ \omega_c \\ q_t \\ \omega_t \end{bmatrix} \quad (۲۷)$$

این بردار وضعیت شامل شش مؤلفه کلیدی است که هرکدام متغیرهای اساسی برای توصیف وضعیت دینامیکی و کینماتیکی سیستم هستند:

۱. موقعیت دنبال‌کننده r در چارچوب LVLH.
۲. سرعت دنبال‌کننده v در چارچوب LVLH.
۳. جهت‌گیری چارچوب بدنه دنبال‌کننده q_c نسبت به چارچوب LVLH.
۴. نرخ چرخش چارچوب بدنه دنبال‌کننده ω_c نسبت به چارچوب LVLH.
۵. جهت‌گیری چارچوب بدنه هدف q_t نسبت به چارچوب LVLH.
۶. نرخ چرخش چارچوب بدنه هدف ω_t نسبت به چارچوب LVLH.

این متغیرها به‌طور جامع وضعیت نسبی و دینامیک چرخشی هر دو جسم (دنبال‌کننده و هدف) را در مأموریت‌های ملاقات فضایی توصیف می‌کنند. سیاست کنترل با هدایت عملگرها، نیروی خالص F_c و گشتاور M_c را بر روی دنبال‌کننده اعمال می‌کند. این نیروها و گشتاورها در امتداد سه محور چارچوب بدنه دنبال‌کننده عمل می‌کنند، زیرا عملگرها به چارچوب بدنه متصل هستند. نیروی خالص اعمال‌شده توسط سیاست کنترل، تغییر لحظه‌ای در سرعت ΔV دنبال‌کننده در ابتدای هر گام زمانی ایجاد می‌کند. اندازه این تغییر سرعت متناسب با طول گام زمانی Δt و معکوس جرم m دنبال‌کننده است.

$$F_c = \begin{bmatrix} F_{x,c} \\ F_{y,c} \\ F_{z,c} \end{bmatrix}, M_c = \begin{bmatrix} M_{x,c} \\ M_{y,c} \\ M_{z,c} \end{bmatrix} \quad (۲۸)$$

$$\begin{bmatrix} r \\ v \end{bmatrix}_{t_0+\Delta t} = \emptyset_{cw} \begin{bmatrix} r \\ v \end{bmatrix}_{t_0} \quad (۲۹)$$

$$\Delta\omega = I^{-1}[M - \omega \times I\omega]\Delta t \quad (۳۰)$$

به‌طور مشابه، گشتاور خالص نیز تغییر لحظه‌ای در نرخ چرخش $\Delta\omega$ ایجاد می‌کند که با استفاده از معادله چرخش اویلر تخمین زده می‌شود

موقعیت و سرعت دنبال‌کننده با گذر زمان طبق مدل کلُهِسی–ویلتیِر (CW) تغییر می‌کنند. برای محاسبه موقعیت و سرعت جدید در هر گام زمانی، از حل تحلیلی توصیف‌شده در معادله زیر استفاده می‌شود:

$$\begin{bmatrix} r \\ v \end{bmatrix}_{t_0+\Delta t} = \emptyset_{cw} \begin{bmatrix} r \\ v \end{bmatrix}_{t_0} \quad (۳۱)$$

تغییر سرعت ΔV اعمال‌شده در ابتدای هر گام زمانی به سرعت اضافه می‌شود:

$$\begin{bmatrix} r \\ v \end{bmatrix}_{t+\Delta t} = \emptyset_{cw} \begin{bmatrix} r \\ v + \Delta V \end{bmatrix}_t \quad (۳۲)$$

جهت‌گیری و نرخ چرخش دنبال‌کننده و هدف با استفاده از انتگرال‌گیری عددی به روش رانگ–کوتا مرتبه چهارم (RK4) تعیین می‌شوند. ورودی‌های تابع RK4 شامل طول گام زمانی، مقادیر فعلی جهت‌گیری و نرخ چرخش، و مشتقات آن‌هاست. تابع RK4 با استفاده از این ورودی‌ها، جهت‌گیری و نرخ چرخش جدید را در پایان گام زمانی محاسبه می‌کند:

$$\begin{bmatrix} q \\ \omega \end{bmatrix}_{t+\Delta t} = RK4\left(\begin{bmatrix} q \\ \omega + \Delta\omega \end{bmatrix}_t, \begin{bmatrix} \dot{q} \\ \dot{\omega} \end{bmatrix}_t, \Delta t\right) \quad (۳۳)$$

این مدل‌سازی دینامیکی به‌طور دقیق رفتار سیستم را در طول مأموریت توصیف می‌کند و امکان کنترل بهینه فضاپیما را فراهم می‌آورد.

۴-۲. پیاده‌سازی

برای سازگاری با چارچوب یادگیری تقویتی فرا آموزشی، مدل سناریوی ملاقات به‌عنوان یک فرآیند تصمیم‌گیری نیمه‌مشاهده‌پذیر (POMDP)

فرمول‌بندی می‌شود. در این چارچوب، POMDP مشابه یک سیستم دینامیکی گسسته عمل می‌کند که وضعیت سیستم آن در پاسخ به خروجی سیاست کنترل در گام‌های زمانی گسسته تکامل می‌یابد. وضعیت کنونی سیستم به‌صورت مشاهداتی که حاوی اطلاعات محدودی هستند، به سیاست کنترل منتقل می‌شود. این مشاهده به‌صورت یک آرایه یک‌بعدی از متغیرهای وضعیت نمایش داده می‌شود، همان‌طور که در معادله (۳۴) نشان داده شده است:

$$Observation = \begin{bmatrix} r \\ v \\ q_c \\ \omega_c \\ q_T \end{bmatrix}_{Norm} \quad (۳۴)$$

این مشاهده ناقص می‌تواند سناریوهایی را نشان دهد که در آن قابلیت‌های داخلی دنبال‌کننده به دلیل عواملی مانند عدم وجود حسگرها یا نقص در تجهیزات محدود شده‌اند. از آنجایی که سیاست کنترل نمی‌تواند به‌طور مستقیم نرخ چرخش هدف ω_T را مشاهده کند، باید از طریق فرا-آموزش به‌طور غیرمستقیم آن را استنباط کند. پس از دریافت یک مشاهده، سیاست کنترل خروجی خود را به POMDP منتقل می‌کند که به‌صورت یک آرایه یک‌بعدی شامل نیروها و گشتاورهای خالص اعمال‌شده توسط عملگرها بر روی دنبال‌کننده بیان می‌شود:

$$Action = \begin{bmatrix} F_{x,c} \\ F_{y,c} \\ F_{z,c} \\ M_{x,c} \\ M_{y,c} \\ M_{z,c} \end{bmatrix}_{Norm} \quad (۳۵)$$

مقادیر مشاهدات و اقدامات در محدوده [-1,1] نرمال‌سازی می‌شوند، همان‌طور که در راهنمای توسعه‌دهنده کتابخانه SB3[14] توصیه شده است. نرمال‌سازی مشاهدات می‌تواند نرخ همگرایی الگوریتم یادگیری را با اطمینان از اینکه گرادیان‌های شبکه دارای مقادیر مشابه هستند، بهبود بخشد [۳۸].

پس از دریافت اقدام از سیاست کنترل، POMDP وضعیت سیستم را از طریق معادلات دینامیکی به‌روزرسانی می‌کند و مشاهده جدیدی از وضعیت محیط و مقدار پاداش متناظر را تولید می‌کند. مشاهدات جدید به سیاست کنترل بازگردانده می‌شوند و این فرآیند به‌صورت چرخه‌ای ادامه می‌یابد. آموزش به قسمت‌هایی تقسیم می‌شود که هر قسمت نمایانگر یک تلاش برای ملاقات است. هر قسمت در صورت اتمام زمان یا فاصله بیش از حد دنبال‌کننده از هدف خاتمه می‌یابد. پس از هر قسمت، MDP به شرایط اولیه بازنشانی می‌شود. این فرآیند با استفاده از زبان پایتون و کتابخانه OpenAI Gym پیاده‌سازی شده است. OpenAI Gym یک چارچوب استاندارد برای توسعه و آزمایش الگوریتم‌های یادگیری تقویتی فراهم می‌کند و امکان ایجاد محیط‌های سفارشی برای آموزش و ارزیابی سیاست کنترل را می‌دهد. این پیاده‌سازی به‌طور خاص برای سناریوهای ملاقات فضایی طراحی شده است و از قابلیت‌های یادگیری تقویتی برای بهبود عملکرد سیاست کنترل استفاده می‌کند. این رویکرد با ترکیب مدل‌سازی دینامیکی، نرمال‌سازی داده‌ها و استفاده از چارچوب‌های پیشرفته یادگیری تقویتی، امکان آموزش مؤثر و کارآمد سیاست‌های کنترل را فراهم می‌آورد.

۴-۳. تابع پاداش

تابع پاداش نقش کلیدی در تعیین رفتار کنترلر آموزش‌دیده ایفا می‌کند. از آنجایی که الگوریتم یادگیری تقویتی تلاش می‌کند تا پاداش تجمعی را به حداکثر برساند، طراحی تابع پاداش باید به گونه‌ای باشد که سیاست کنترل را برای دستیابی به اهداف سناریو تشویق کند. در سناریوی مرحله نهایی ملاقات، هدف اصلی این است که فضاپیمای

دنبال‌کننده شرایط پایانی لازم برای آغاز فاز الحاق را برآورده کند. یک رویکرد ساده این است که هنگام رسیدن به این شرایط پایانی، پاداش مثبتی به سیاست کنترل داده شود و در غیر این صورت هیچ پاداشی تعلق نگیرد، همان‌طور که در معادله (۳۶) نشان داده شده است:

$$R_* = \begin{cases} 0 \\ k_* \end{cases} \quad (۳۶)$$

در این معادله، k_* یک ثابت مثبت است.

مشکل این رویکرد آن است که به یک تابع پاداش پراکنده منجر می‌شود. به عبارت دیگر، تنها بخش بسیار کوچکی از فضای حالت محیط، پاداشی به سیاست کنترل می‌دهد، در حالی که بقیه فضای حالت هیچ بازخوردی ارائه نمی‌کنند. با یک تابع پاداش پراکنده، سیاست کنترل ممکن است مدت زمان زیادی برای بهبود نیاز داشته باشد، زیرا فقط به اطلاعات محدودی دسترسی دارد که بتواند از آنها یاد بگیرد. در این وضعیت، سیاست کنترل مجبور است فضای حالت را بدون هیچ‌گونه راهنمایی پاداشی کاوش کند تا در نهایت شرایط پایانی را بیابد.

برای حل مشکل پاداش پراکنده، می‌توان تابع پاداش را با استفاده از دانش تخصصی به گونه‌ای طراحی کرد که پاداش‌های کوچکتر و میان‌مرحله‌ای به سیاست کنترل داده شود و آن را به هدف اصلی هدایت کند. این پاداش‌های کمکی می‌توانند رفتارهایی که به دستیابی به هدف منجر می‌شوند را تشویق کنند و در عین حال رفتارهای نادرست را جریمه کنند. برای ایجاد چنین تابعی، هدف کلی سناریو به چندین هدف کوچکتر تقسیم می‌شود. به عنوان مثال:

۱. نشانه‌گیری دقیق دنبال‌کننده به سمت هدف.
۲. جلوگیری از برخورد با هدف یا ناحیه ممنوعه.
۳. کاهش مصرف سوخت توسط دنبال‌کننده.

تابع پاداش نهایی شامل چندین بخش می‌شود که هر یک به یکی از اهداف کوچکتر اختصاص دارد و بخش آخر همان پاداش پراکنده اولیه است:

$$R = R_\theta + R_f + R_c + R_* \quad (۳۷)$$

هدف از پاداش نشانه‌گیری R_θ این است که دنبال‌کننده در طول مسیر، همواره به سمت هدف نشانه‌گیری شده باقی بماند. خطای نشانه‌گیری θ_e به صورت زاویه‌ای بین جهت کنونی نشانه‌گیری دنبال‌کننده و جهت مطلوب نشانه‌گیری اندازه‌گیری می‌شود. مقدار $\theta_{e,max}$ بیشترین خطای مجاز نشانه‌گیری است. اگر خطای نشانه‌گیری از این مقدار فراتر رود، مرحله زودتر از موعد خاتمه می‌یابد. پاداش نشانه‌گیری به صورت زیر تعریف می‌شود:

$$R_\theta = k_\theta \left(1 - \frac{\theta_e}{\theta_{e,max}}\right) \quad (۳۸)$$

پاداش مصرف سوخت R_f سیاست کنترل را به تولید مسیرهایی با مصرف بهینه سوخت تشویق می‌کند. مقدار سوخت مصرف‌شده در هر گام زمانی با توجه به تغییر سرعت ΔV اعمال‌شده بر دنبال‌کننده تخمین زده می‌شود. این پاداش به صورت خطی تنظیم شده است؛ به طوری که اگر در یک گام زمانی ΔV حداکثر مقدار خود باشد، پاداشی به عامل داده نمی‌شود، اما در گام‌هایی که هیچ ΔV اعمال نمی‌شود، عامل پاداش k_f دریافت می‌کند.

$$R_f = k_f \left(1 - \frac{|\Delta V|}{\Delta V_{max}}\right) \quad (۳۹)$$

پاداش جلوگیری از برخورد R_c برای جریمه کردن عامل در صورت وقوع برخورد طراحی شده است. در هر گام زمانی که دنبال‌کننده در ناحیه ممنوعه قرار گیرد، جریمه ثابتی به مقدار k_c اعمال می‌شود.

$$R_c = \begin{cases} 0 \\ -k_c \end{cases} \quad (۴۰)$$

یک نکته مهم در طراحی تابع پاداش این است که فضای حالت موجود با گذر زمان کاهش می‌یابد. در ابتدای هر اپیزود، به دنبال‌کننده اجازه داده می‌شود تا حداکثر فاصله d_{max} از هدف دور شود. اما با پیشرفت اپیزود، مقدار d_{max} به تدریج کاهش می‌یابد و فضایی که دنبال‌کننده می‌تواند در آن حرکت کند، محدودتر می‌شود. اگر دنبال‌کننده از d_{max} عبور کند، اپیزود جاری متوقف می‌شود. این کاهش تدریجی مقدار d_{max} ادامه دارد تا زمانی که به لبه ناحیه ممنوعه برسد، بنابراین دنبال‌کننده زمان بیشتری برای کاوش در فضای نزدیکتر به هدف دارد. این روند به صورت ضمنی دنبال‌کننده را تشویق می‌کند تا بدون نیاز به افزودن یک مؤلفه موقعیتی به تابع پاداش، به هدف نزدیکتر شود. این طراحی تابع پاداش با ترکیب پاداش‌های کمکی و جریمه‌ها، به سیاست کنترل کمک می‌کند تا به‌طور مؤثرتری به اهداف سناریو دست یابد و از مشکلات ناشی از پاداش‌های پراکنده جلوگیری کند.

۵. ارزیابی عملکرد

این بخش به توصیف نحوه آموزش و ارزیابی سیاست‌های کنترل می‌پردازد. ابتدا تنظیمات و نتایج فرآیند آموزش هر دو سیاست کنترل ارائه می‌شود. سپس، عملکرد سیاست‌ها در سناریوهای اصلی از نظر خطاهای حالت پایانی، برخوردها و مصرف سوخت ارزیابی می‌شود. در نهایت، همان معیارهای عملکرد برای ۱۰۰۰ سناریوی تصادفی بررسی می‌شوند.

۵-۱. آموزش

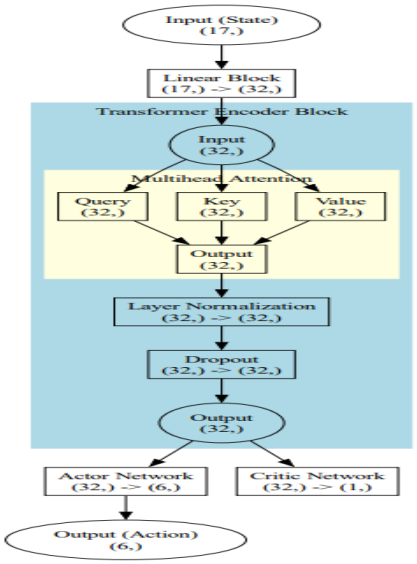
دو شبکه عصبی آموزش داده شدند :یکی با استفاده از سیاست بازگشتی و دیگری با استفاده از سیاست ترنسفورمرها. در طول آموزش، عملکرد سیاست‌ها به‌طور دوره‌ای ارزیابی می‌شود. این تابع قبل از هر به‌روزرسانی سیاست اجرا می‌شود و میانگین پاداش تجمعی سیاست را در طول پنجاه اپیزود محاسبه می‌کند، که هر اپیزود با شرایط اولیه تصادفی آغاز می‌شود. برای شبیه‌سازی سناریوهای دارای عدم قطعیت، سیاست‌ها فقط یک مشاهده نسبی از محیط دریافت می‌کنند که شامل تمامی متغیرهای حالت به جز نرخ چرخش هدف است. برای جلوگیری از بیش‌برازش سیاست به شرایط خاص، هر اپیزود آموزش با شرایط اولیه‌ای که به‌طور تصادفی انتخاب شده‌اند، آغاز می‌شود. محدوده شرایط اولیه برای هر متغیر حالت همراه با مقادیر اسمی در جدول ۲ نشان داده شده است. وضعیت اولیه اسمی دنبال‌کننده در نقطه‌ای در فاصله ۲۰ متری از هدف در امتداد محور y قرار دارد. در حالت اولیه اسمی، دنبال‌کننده به سمت هدف نشانه‌گیری شده و راهرو ورودی هدف نیز به سمت دنبال‌کننده قرار دارد. در آغاز هر اپیزود، شرایط اولیه از یک توزیع یکنواخت تصادفی حول مقدار اسمی نمونه برداری می‌شود. تغییر شرایط اولیه کمک می‌کند تا شبکه‌ها به مجموعه‌ای خاص از شرایط اولیه محدود نشوند و استحکام سیاست‌ها را افزایش دهد. شبکه پیشنهادی در ترنسفورمر مطابق شکل ۳ است که تعداد پارامترهای شبکه ترانسفرمر ۵۰۹۵ است در حالیکه تعداد پارامترهای شبکه LSTM برابر ۶۰۴۷۷۸ است.

جدول ۲- شرایط اولیه سناریو

پارامتر	مقدار نامی	بازه تغییرات
r	20m	$[15-25]$
v	0 m/s	$[0-0.5]$
q_c	0 deg	$[0-1]$
ω_c	0 deg/s	$[0-.2]$
q_T	0 deg	$[0-45]$
ω_T	0 deg/s	$[0-3]$

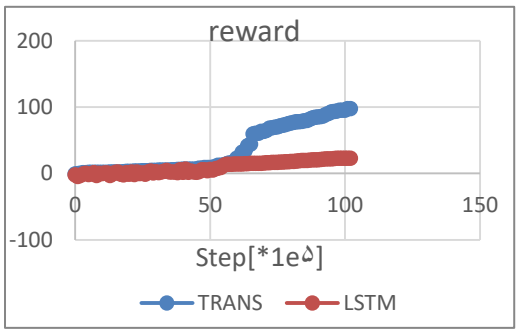
شکل ۳ بلوک دیاگرام شبکه ترنسفورمر ساده شده پیشنهادی را نشان داده است. این مدل شامل چندین لایه با عملکردهای مشخص است. لایه ورودی، داده‌های ورودی با ابعاد ۱۷ (معادل تعداد حالت‌های محیط) را به ۳۲ ویژگی تبدیل می‌کند.

سپس، داده‌ها وارد لایه رمزگذار می‌شوند که با استفاده از یک لایه سفارشی به نام CustomTransformerEncoderLayer طراحی شده است. این لایه از مکانیزم توجه چندسری برای یادگیری روابط و وابستگی‌های بین بخش‌های مختلف داده استفاده می‌کند. علاوه بر این، لایه NonDynamicallyQuantizableLinear خروجی‌های ۳۲ ویژگی را پردازش کرده و به همان تعداد ویژگی بازمی‌گرداند. برای بهبود پایداری فرآیند یادگیری، از لایه نرمال‌سازی (LayerNorm) استفاده شده است و لایه Dropout نیز به منظور جلوگیری از بیش‌برازش و بهبود قابلیت عمومی‌سازی مدل به کار گرفته شده است. در انتهای شبکه، لایه‌های Actor و Critic قرار دارند. لایه Actor، خروجی ۳۲ ویژگی را به ۶ ویژگی تبدیل می‌کند که نشان‌دهنده اقدامات ممکن مدل در محیط هستند. لایه Critic نیز خروجی ۳۲ ویژگی را به یک ویژگی کاهش می‌دهد که مقدار تابع ارزش را نشان می‌دهد. این طراحی به‌طور خاص برای پردازش داده‌های پیچیده محیط و تولید خروجی‌های دقیق در مسائل تصمیم‌گیری متوالی بهینه‌سازی شده است.



شکل ۳- نمایش بلوک دیاگرام شبکه ترنسفورمر ساده شده پیشنهادی

شکل ۴ یک نمودار مقایسه‌ای است که عملکرد دو مدل مختلف را در طول مراحل یادگیری نشان می‌دهد. محور افقی تعداد مراحل (Step) را در مقیاس 1e5 و محور عمودی میزان پاداش[۱۵] را نمایش می‌دهد. خط آبی نمایانگر عملکرد مدل مبتنی بر ترنسفورمر است. همان‌طور که مشاهده می‌شود، این مدل پس از حدود ۵۰ مرحله بهبود چشمگیری در میزان پاداش دارد و پاداش آن به بیش از ۱۰۰ می‌رسد. این امر نشان‌دهنده قدرت این مدل در یادگیری و انطباق با محیط در طول مراحل یادگیری است. خط قرمز عملکرد مدل LSTM را نمایش می‌دهد. این مدل رشد کمتری در میزان پاداش دارد و به حداکثر حدود ۲۰ می‌رسد. این نتیجه نشان می‌دهد که مدل LSTM در مقایسه با مدل ترنسفورمر عملکرد ضعیف‌تری داشته و کمتر توانسته است به حداکثر پاداش دست یابد. پاداش میانگین معیار مفیدی برای تجسم فرآیند یادگیری است، اما شاخص کاملی برای ارزیابی عملکرد کلی سیاست‌ها محسوب نمی‌شود. در بخش‌های بعدی، عملکرد سیاست‌ها با استفاده از معیارهای بصری‌تر و دقیق‌تر ارزیابی خواهد شد.



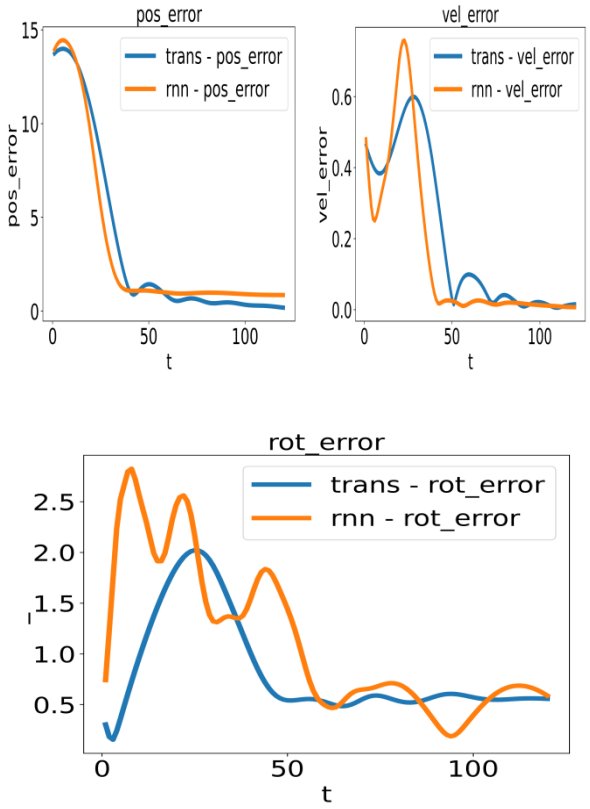
شکل ٤- مقایسه نمودارهای یادگیری دو سیاست

۵-۲. عملکرد در سناریوی اسمی

در سناریوی اسمی، شرایط اولیه نمایانگر ساده‌ترین حالت نزدیک‌شدن نهایی است که سیاست کنترل ممکن است با آن مواجه شود. در این سناریو، رهگیر ابتدا به‌صورت ایستا در یک نقطه‌ی نگهداری

قرار دارد که ۲۰ متر با هدف فاصله دارد و مستقیم به سمت هدف اشاره کرده است. برای ارزیابی مسیرهای به‌دست‌آمده، از چهار شاخص اصلی استفاده می‌شود: خطای موقعیت نهایی r_e، خطای سرعت نهایی v_e، خطای نرخ چرخش نهایی ω_e و ΔV کل که بهره‌وری سوخت را اندازه‌گیری می‌کند. شکل ۵ واکنش حلقه‌بسته سیاست‌های بازگشتی (LSTM) و ترنسفورمر را زمانی که سناریو با شرایط اسمی آغاز می‌شود، نشان می‌دهد. مشاهده می‌شود که سیاست‌ها در رفتار موقعیت و سرعت مشابه هستند. پس از تقریباً ۴۰ ثانیه، هر دو سیاست قید موقعیت نهایی را برآورده می‌کنند و پس از ۴۵ ثانیه، قید سرعت را نیز برآورده کرده و با موفقیت به نقطه‌ی مطلوب برای الحاق می‌رسند و در ادامه‌ی مسیر در آنجا باقی می‌مانند.

شکل ۵ شامل سه نمودار مجزا است که خطاهای موقعیت (pos_error)، سرعت (vel_error) و نرخ چرخش (rot_error) را در دو مدل مختلف مورد ارزیابی و مقایسه قرار می‌دهد. در نمودار خطای موقعیت، هر دو مدل در ابتدا خطای مشابهی دارند، اما به مرور زمان خطا کاهش می‌یابد. مدل ترنسفورمر در کاهش خطای موقعیت کمی بهتر از مدل LSTM عمل می‌کند، زیرا خطای آن در مراحل اولیه سریع‌تر کاهش می‌یابد و به مقدار کمتری می‌رسد. در نمودار خطای سرعت، هر دو مدل در ابتدا خطای مشابهی دارند، اما مدل ترنسفورمر نسبت به مدل LSTM سریع‌تر به سمت خطای صفر حرکت می‌کند و در زمان کمتری به خطای پایین‌تری می‌رسد. در نمودار خطای نرخ چرخش، مدل ترنسفورمر در ابتدا خطای بیشتری نسبت به LSTM دارد، اما به مرور زمان کاهش یافته و به خطای کمتری نسبت به LSTM می‌رسد. این نشان‌دهنده‌ی این است که مدل ترنسفورمر در بلندمدت عملکرد بهتری در کاهش خطای چرخش دارد. با توجه به این نمودارها، مدل ترنسفورمر در تمام خطاهای موقعیت، سرعت و نرخ چرخش عملکرد بهتری نسبت به مدل LSTM داشته و توانسته این خطاها را سریع‌تر و با مقدار نهایی کمتری کاهش دهد.



شکل ۵- خطاهای موقعیت ، سرعت و نرخ چرخش دو سیاست

جدول ۳ معیارهای عملکرد دو مسیر را نشان می‌دهد. میانگین خطاها $(\theta_e , \omega_e , v_e , r_e)$ از زمانی که قیدهای موقعیت و سرعت برآورده می‌شوند تا انتهای مسیر محاسبه شده‌اند. داده‌های جدول به وضوح نشان می‌دهند که سیاست ترنسفورمر در اغلب پارامترهای کلیدی، عملکرد بهتری نسبت به سیاست LSTM داشته است، به‌ویژه در خطای فاصله و زاویه و همچنین کاهش مصرف انرژی برای تغییرات سرعت زاویه‌ای. این نتایج نشان می‌دهند که مدل ترنسفورمر نه‌تنها در کاهش خطاها عملکرد بهتری دارد، بلکه در مصرف سوخت نیز بهینه‌تر عمل می‌کند. این بهبود

عملکرد، مدل ترنسفورمر را به‌عنوان گزینه‌ای برتر برای سناریوهای ملاقات مداری معرفی می‌کند.

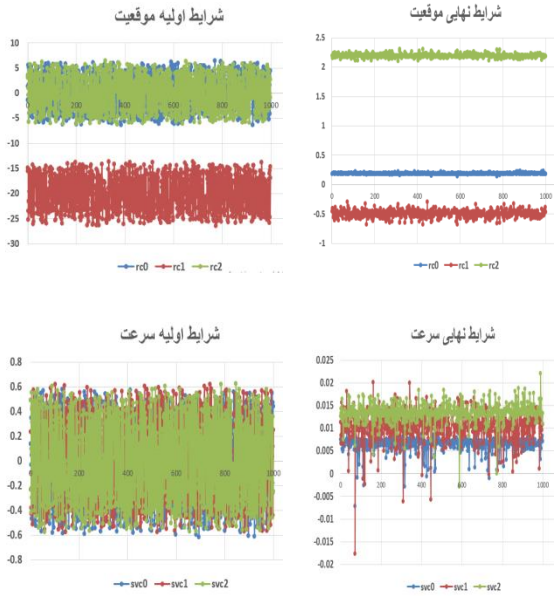
جدول ۳- معیارهای عملکرد مسیر نامی برای دو سیاست

پارامتر	واحد	LSTM	ترنسفورمر
میانگین r_e	m	0.85	0.2
میانگین v_e	m/s	0.005	0.01
میانگین θ_e	deg	1.68	0.53
میانگین ω_e	deg/s	0.58	0.51
کل Δv	m/s	0.024	0.02
کل $\Delta \omega$	deg/s	0.13	0.057

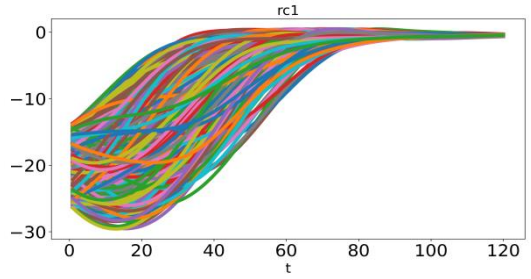
۵-۳. شبیه‌سازی مونت‌کارلو

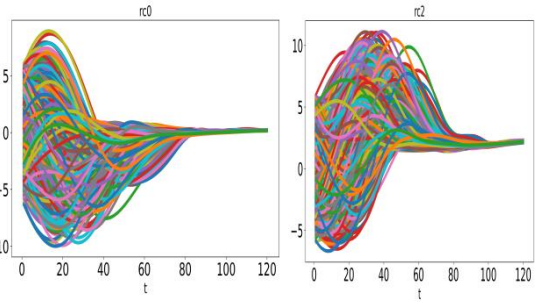
شبیه‌سازی‌های مونت‌کارلو به‌عنوان روش‌های تأیید مبتنی بر آزمایش، اغلب برای ارزیابی قابلیت اطمینان و استحکام سیاست‌های شبکه عصبی استفاده می‌شوند [۱۲, ۳۷, ۳۹]. هدف از این نوع شبیه‌سازی، ارزیابی عملکرد یک سیاست در طیف گسترده‌ای از شرایط اولیه تصادفی نمونه‌برداری شده است. پس از نمونه‌گیری کافی، می‌توان نمای کلی از محدوده‌ی نتایج ممکن به دست آورد. شبیه‌سازی مونت‌کارلو به تنهایی نمی‌تواند ثابت کند که یک سیاست عصبی همیشه مطابق انتظار رفتار خواهد کرد، اما می‌تواند ابزاری مفید برای تعیین محدودیت‌های سیاست و شناسایی موارد خاصی که در آن عملکرد ضعیفی دارد، باشد. برای این شبیه‌سازی مونت‌کارلو، ۱۰۰۰ شرایط اولیه از همان توزیع تصادفی که در طول فرآیند آموزش به کار گرفته شده بود، نمونه‌برداری شد. هر دو سیاست بازگشتی (LSTM) و ترنسفورمر برای تولید مسیرها از این شرایط اولیه استفاده شدند. همان معیارهای مورد استفاده در آزمایش‌های سناریوی اسمی نیز در این شبیه‌سازی ثبت شد.

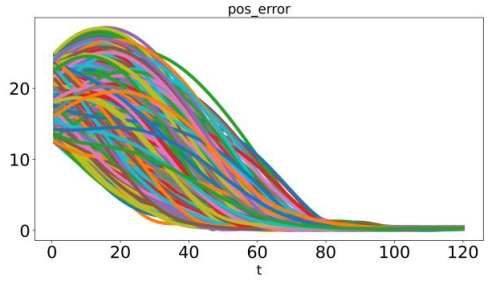
شکل ۶ شامل چهار نمودار است که شرایط اولیه و نهایی اجرای مونت‌کارلو برای سیاست ترنسفورمر را نمایش می‌دهد. نمودار بالا سمت چپ توزیع مقادیر اولیه موقعیت را در سه محور فضایی x، y و z نشان می‌دهد. مقادیر اولیه موقعیت برای هر محور به‌صورت پراکنده و در یک بازه معین مطابق جدول ۲ توزیع شده‌اند. نمودار بالا سمت راست شرایط موقعیت فضاپیما را در زمان‌های پایانی مونت‌کارلو برای موقعیت‌های موفق نمایش می‌دهد. در این نمودار، پراکندگی داده‌ها نسبت به نمودار شرایط اولیه کمتر است که نشان می‌دهد فضاپیما به موقعیت پایدار و موفقیت‌آمیز در پایان مسیر دست یافته است. نمودار پایین سمت چپ شرایط اولیه سرعت را در سه محور فضایی v_x، v_y و v_z نمایش می‌دهد که با رنگ‌های متفاوت مشخص شده‌اند. مقادیر سرعت اولیه در هر محور به‌صورت نقاط پراکنده در یک بازه محدود نمایش داده شده‌اند که نشان‌دهنده توزیع تصادفی سرعت در آغاز شبیه‌سازی مطابق جدول ۲ است. نمودار پایین سمت راست شرایط پایانی سرعت را در مونت‌کارلو برای موقعیت‌های موفقیت‌آمیز نشان می‌دهد. مقادیر نهایی سرعت نشان می‌دهد که نوسانات سرعت در شرایط پایانی نسبت به شرایط اولیه کمتر است، که بیانگر نزدیک شدن فضاپیما به شرایط مطلوب و کنترل شده است. این نمودارها به‌طور کلی توزیع مقادیر موقعیت و سرعت در شرایط اولیه و نهایی را برای سیاست ترنسفورمر نشان می‌دهند. نمودارهای پایانی (سمت راست) بیانگر بهبود پایداری و دقت سیستم در اجرای مانورهای مداری و موفقیت در شرایط مختلف مونت‌کارلو هستند، به‌طوری‌که موقعیت و سرعت به مقادیر پایدار و محدودی رسیده‌اند که نشان‌دهنده موفقیت و کنترل بهینه سیستم است.



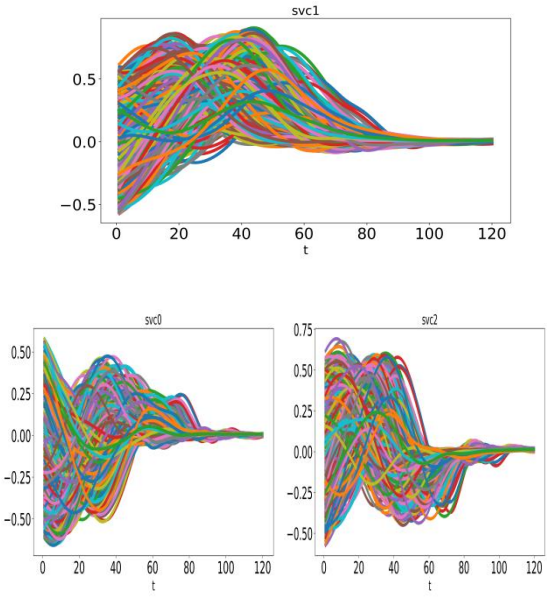
شکل ۶- شرایط اولیه و نهایی اجرای مونت کارلو (ترنسفورمر)

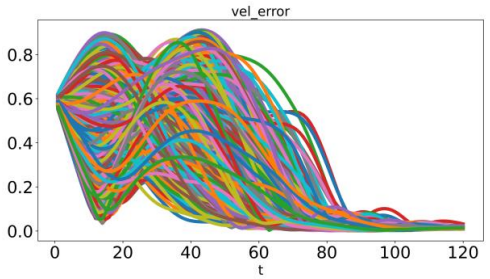






شکل ۷- نتایج اجرای مونت کارلو موقعیت و خطای آن (ترنسفورمر)





شکل ۸- نتایج اجرای مونت کارلو سرعت و خطای آن (ترنسفورمر)

نمودارهای شکل ۷ و ۸ به‌طور کلی نشان‌دهنده عملکرد موفقیت‌آمیز سیاست ترنسفورمر در کاهش سریع خطا و پارامترهای موقعیت و سرعت در فرآیند کنترل و هدایت هستند. هر کدام از متغیرها، پس از گذشت زمان کوتاهی از شروع فرآیند، به حالت پایداری می‌رسند و نوسانات آن‌ها کاهش می‌یابد.

جدول ٤- معیارهای عملکرد مسیرهای اجرای مونت کارلو

پارامتر	واحد	LSTM	ترنسفورمر
میانگین r_e	m	0.8	0.3
میانگین v_e	m/s	0.008	0.014
میانگین θ_e	deg	2.17	1.05
میانگین ω_e	deg/s	0.51	0.48
کل Δv	m/s	2.1	1.98
کل $\Delta \omega$	deg/s	0.3	0.021

جدول ۴ معیارهای عملکرد مسیرهای اجرای مونت‌کارلو را برای دو سیاست کنترلی مختلف، یعنی LSTM و ترنسفورمر، مقایسه می‌کند. نتایج نشان می‌دهد که مدل ترنسفورمر در برخی معیارها مانند میانگین فاصله خطای موقعیتی و خطای زاویه‌ای عملکرد بهتری نسبت به LSTM دارد، در حالی که LSTM در کاهش خطای سرعت عملکرد بهتری از خود نشان می‌دهد. همچنین، مجموع تغییرات سرعت و سرعت زاویه‌ای نیز بین دو مدل مقایسه شده‌اند که بیانگر برتری نسبی ترنسفورمر در کنترل تغییرات سرعت زاویه‌ای است. این نتایج نشان می‌دهند که مدل ترنسفورمر در اغلب معیارهای کلیدی عملکرد بهتری دارد و می‌تواند به‌عنوان گزینه‌ای برتر برای سناریوهای ملاقات مداری مورد استفاده قرار گیرد.

۶. جمع‌بندی و نتیجه‌گیری

این مقاله به مقایسه دو رویکرد مبتنی بر شبکه‌های عصبی LSTM و ترنسفورمرها در کنترل بهینه مأموریت‌های ملاقات مداری پرداخت. شبکه‌ها با سیاست‌های بازگشتی و ترنسفورمرها آموزش داده شدند و عملکرد آن‌ها در شرایط اسمی و شبیه‌سازی‌های مونت‌کارلو ارزیابی گردید. نتایج نشان داد که مدل ترنسفورمر در کاهش خطاهای موقعیت، سرعت و نرخ چرخش عملکرد بهتری دارد و با دقت بیشتری به شرایط مطلوب برای الحاق می‌رسد. همچنین، این مدل در کاهش مصرف انرژی و تغییرات سرعت زاویه‌ای برتری قابل توجهی از خود نشان داد. این برتری ناشی از توانایی ترنسفورمرها در یادگیری روابط پیچیده و وابستگی‌های بلندمدت در داده‌هاست، که آن‌ها را به گزینه‌ای ایده‌آل برای کنترل بهینه در محیط‌های پویا و نامطمئن تبدیل می‌کند. شبکه ترنسفورمر پیشنهادی، با کاهش ۳۰ درصدی خطای موقعیت و مصرف انرژی، عملکردی برتر نسبت به LSTM داشت. این مدل با تنها ۵۰۹۵ پارامتر در مقایسه با ۶۰۴۷۷۸ پارامتر LSTM، نه تنها دقت و پایداری سیستم را بهبود بخشید، بلکه امکان پیاده‌سازی عملیاتی بر روی سخت‌افزارهای موجود را نیز فراهم کرد. شبیه‌سازی‌های مونت‌کارلو تأیید کرد که ترنسفورمرها در مواجهه با عدم‌قطعیت‌ها، قابلیت اطمینان و تطبیق‌پذیری بالایی دارند. این مدل توانست در طیف گسترده‌ای از شرایط اولیه، خطاهای موقعیت و سرعت را به‌طور مؤثر کاهش دهد و به شرایط پایدار برسد، در حالی که LSTM نوسانات بیشتری در خطاها و مصرف انرژی نشان داد. این یافته‌ها تأیید می‌کند که ترنسفورمرها نه تنها دقت و کارایی سیستم را بهبود می‌بخشند، بلکه قابلیت اطمینان آن را در شرایط مختلف افزایش می‌دهند.

این پژوهش گامی مهم در توسعه سیستم‌های هدایت و کنترل خودمختار برای مأموریت‌های فضایی آینده، از جمله سرویس‌دهی و جمع‌آوری زباله‌های مداری، محسوب می‌شود. نتایج نشان می‌دهد که ترنسفورمرها ابزاری مؤثر برای تحقق سامانه‌های هوشمند و انعطاف‌پذیر در سناریوهای ملاقات مداری شش درجه آزادی هستند.

۷. مراجع

[۱] A. Richards, T. Schouwenaars, J. P. How, and E. Feron, "Spacecraft Trajectory Planning with Avoidance Constraints Using Mixed-Integer Linear Programming," Journal of Guidance, Control, and Dynamics, vol. 25, no. 4, pp. 755-764, 2002/07/01 2002, doi: ۱.۲,٤٩٤٣/۰,۲٥۱٤

[۲] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge: Cambridge University Press, 2004.

[۳] E. Hartley, A tutorial on model predictive control for spacecraft rendezvous. 2015, pp. 1355-1361.

[٤] D. Woffinden and D. Geller, "Optimal Orbital Rendezvous Maneuvering for Angles-Only Navigation," Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM, vol. 32, pp. 1382-1387, 07/01 2009, doi: 10.2514/1.45006.

[٥] L. Blackmore and D. Scharf, "Minimum-Landing-Error Powered-Descent Guidance for Mars Landing Using Convex

Optimization," Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM, vol. 33, pp. 1161-1171, 07/01 2010, doi: 10.2514/1.47202.
[٦] L. Blackmore, "Autonomous precision landing of space rockets," vol. 46 ,pp. 15-20, 01/01 2016.
[٧] U. Eren, A. Prach, B. B. Koçer, S. V. Raković, E. Kayacan, and B. Açıkmeşe, "Model Predictive Control in Aerospace Systems: Current State and Opportunities," Journal of Guidance, Control, and Dynamics, vol. 40, no. 7, pp. 1541-1 ٠١/٠٧/٢٠١٧ ,٥٦٦ ,٢٠١٧doi: 10.2514/1.G002507.
[٨] B. Benedikter, A. Zavoli, G. Colasurdo, S. Pizzurro, and E. Cavallini, "Autonomous Upper Stage Guidance Using Convex Optimization and Model Predictive Control," in ASCEND 2020, (ASCEND: American Institute of Aeronautics and Astronautics, 2020.
[٩] D. Miller, J. A. Englander, and R. Linares, "Interplanetary low-thrust design using proximal policy optimization," in 2019 AAS/AIAA Astrodynamics Specialist Conference, 2019, no. GSFC-E-DAA-TN71225 .
[١٠] A. Zavoli and L. Federici, "Reinforcement Learning for Robust Trajectory Design of Interplanetary Missions," Journal of Guidance, Control, and Dynamics, vol. 44, no. 8, pp. 1440-1453, 2021, doi: 10.2514/1.G005794.
[١١] C. J. Sullivan and N. Bosanac, "Using Reinforcement Learning to Design a Low-Thrust Approach into a Periodic Orbit in a Multi-Body System," in AIAA Scitech 2020 Forum, (AIAA SciTech Forum: American Institute of Aeronautics and Astronautics, 2020.
[١٢] N. B. LaFarge, D. Miller, K. C. Howell, and R .Linares, "Guidance for Closed-Loop Transfers using Reinforcement Learning with Application to Libration Point Orbits," in AIAA Scitech 2020 Forum, (AIAA SciTech Forum: American Institute of Aeronautics and Astronautics, 2020.
[١٣] L. Federici, A. Scorsoglio, A. Zavoli, and R. Furfaro, "Autonomous guidance for cislunar orbit transfers via reinforcement learning," in AAS/AIAA Astrodynamics Specialist Conference, 2021: American Astronautical Society Big Sky, Montana (Virtual .(
[١٤] R. Furfaro, A. Scorsoglio, R. Linares, and M. Massari, "Adaptive generalized ZEM-ZEV feedback guidance for planetary landing via a deep reinforcement learning approach," Acta Astronautica, vol. 171, pp. 156-171, 2020/06/01/ 2020, doi: https://doi.org/10.1016/j.actaastro.2020.02 ٠٥١..
[١٥] B. Gaudet, R. Linares, and R. Furfaro, "Deep reinforcement learning for six degree-of-freedom planetary powered descent and landing," arXiv preprint arXiv:1810.08719, 2018.
[١٦] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," arXiv preprint arXiv:1707.06347, 2017.
[١٧] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," arXiv e-prints, p. arXiv:1509.02971, 2015, doi: 10.48550/arXiv.1509.02971.
[١٨] C. E. Oestreich, R. Linares, and R. Gondhalekar, "Autonomous six-degree-of-freedom spacecraft docking maneuvers via reinforcement learning," arXiv preprint arXiv:2008.03215, 2020.
[١٩] L. Federici, A. Scorsoglio, A. Zavoli, and R. Furfaro, "Meta-reinforcement learning for adaptive spacecraft guidance during finite-thrust rendezvous missions," Acta Astronautica, vol. 201, pp. 129-141, 2022/12/01/ 2022, doi: https://doi.org/10.1016/j.actaastro.2022.08.047.
[٢٠] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, "A simple neural attentive meta-learner," arXiv preprint arXiv:1707.03141, 2017.
[٢١] L. C. Melo, "Transformers are meta-reinforcement learners," in international conference on machine learning, 2022: PMLR, pp. 15340-15359 .
[٢٢] J. Ventura, M. Ciarcià, M. Romano, and U. Walter, "Fast and Near-Optimal Guidance for Docking to Uncontrolled Spacecraft," Journal of Guidance, Control, and Dynamics, vol. 40, no. 12, pp. 3138-3154, 2017/12/01 2016, doi: 10.2514/1.G001843.
[٢٣] G. Boyarko, O. Yakimenko, and M. Romano" ,Optimal Rendezvous Trajectories of a Controlled Spacecraft and a Tumbling Object," Journal of Guidance, Control, and Dynamics, vol. 34, no. 4, pp. 1239-1252, 2011/07/01 2011, doi: 10.2514/1.47645.
[٢٤] K. J. Åström, "Optimal control of Markov processes with incomplete state information," Journal of Mathematical Analysis and Applications, vol. 10, pp. 174-205, 1965.
[٢٥] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in International conference on machine learning, 2015: PMLR, pp. 1889-1897 .
[٢٦] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement

Learning with a Stochastic Actor," presented at the Proceedings of the 35th International Conference on Machine Learning, Proceedings of Machine Learning Research, 2018. [Online]. Available: https://proceedings.mlr.press/v80/haarnoja18b.html.
[۲۷] S. Silvestrini et al., "Chapter Fifteen - Modern Spacecraft GNC," in Modern Spacecraft Guidance, Navigation, and Control, V. Pesce, A. Colagrossi, and S. Silvestrini Eds.: Elsevier, 2023, pp. 819-981.
[۲۸] A. Brandonisio and M. Lavagna, Sensitivity Analysis of Adaptive Guidance via Deep Reinforcement Learning for Uncooperative Space Objects Smart Imaging. 202.۱
[۲۹] L. Capra, A. Brandonisio, and M. Lavagna, "Network architecture and action space analysis for deep reinforcement learning towards spacecraft autonomous guidance," Advances in Space Research, vol. 71, 12/01 2022, doi: 10.1016/j.asr.2022.11.048.
[۳۰] T. Wolf et al., Transformers: State-of-the-Art Natural Language Processing. 2020, pp. 38-45.
[۳۱] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in North American Chapter of the Association for Computational Linguistics, 2019 .
[۳۲] H. Chen et al., "Pre-Trained Image Processing Transformer," in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 20-25 June 2021 2021, pp. 12294-12305, doi: 10.11/۰۹CVPR46437.2021.01212 .
[۳۳] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.
[۳٤] A. Vaswani et al., "Attention Is All You Need," 06/12 2017, doi: 10.48550/arXiv.1706.0.۳۷٦۲
[۳٥] M.-T. Luong, "Effective approaches to attention-based neural machine translation," arXiv preprint arXiv:1508.04025, 2015.
[۳٦] S. Hochreiter, "Long Short-term Memory," Neural Computation MIT-Press, 1997.
[۳۷] L. Federici, A. Scorsoglio, A. Zavoli, and R. Furfaro, "Meta-Reinforcement Learning for Adaptive Spacecraft Guidance during Multi-Target Missions," in IAF Astrodynamics Symposium 2021 at the 72nd International Astronautical Congress, IAC 2021, 2021: International Astronautical Federation ,IAF .
[۳۸] S.-B. (SB3). "Reinforcement Learning Tips and Tricks." https://stable-baselines3.readthedocs.io/en/master/guide/rl_tips.html (accessed.
[۳۹] B. Gaudet, R. Linares, and R. Furfaro, "Deep reinforcement learning for six degree-of-freedom planetary landing," Advances in Space Research, vol. 65, no. 7, pp. 1723-1741, 2020/04/01/ 2020, doi: https://doi.org/10.1016/j.asr.2019.12.030.

۸. پی نوشت

1 meta-RL
2 Clohessy–Wiltshire–Hill
3 Chaser
4 Target
5 Belief-space
6 Proximal Policy Optimization
7 Clipped objective function
8 Advantage Function
9 Value Function
10 Entropy Regularization Term
11 Feed-forward neural networks
12 Self-Attention
13 Multi-Head Attention
14 Stable-Baselines3
15 Reward