Contents lists available at ScienceDirect

Computers in Human Behavior

journal homepage: www.elsevier.com/locate/comphumbeh

Advancing spacecraft rendezvous and docking through safety reinforcement learning and ubiquitous learning principles

Kanta Prasad Sharma[a], Indradeep Kumar[b], Pavitar Parkash Singh[c], K. Anbazhagan[d], Hussain Mobarak Albarakati[e], Mohammed Wasim Bhatt[f,*], Avlokulov Anvar Ziyadullayevich[g], Arti Rana[h], Sivasankari S. A[i]

[a] *Department of Computer Engineering and Application, GLA University, Mathura, Uttar Pradesh, 281406, India*
[b] *Department of Aeronautical Engineering, Institute of Aeronautical Engineering, Dundigal, Hyderabad, Telangana, India, 500043*
[c] *Department of Management, Lovely Professional University, Phagwara, India*
[d] *Department of Computer Science and Engineering, Saveetha School of Engineering, SIMATS, Chennai, India*
[e] *Computer Engineering and Networks Department, College of Computer and Information Systems, Umm Al-Qura University, Makkah ,24382, Saudi Arabia*
[f] *Model Institute of Engineering and Technology, Jammu, J&K, India*
[g] *The Department of Audit, Tashkent Institute of Finance, Tashkent, Uzbekistan*
[h] *Department of Computer Science & Engineering, Uttaranchal Institute of Technology, Uttaranchal University, Dehradun, 248007, India*
[i] *Department of ECE, Vignan's Foundation for Science, Technology & Research (Deemed to be University), Vadlamudi, Guntur, 522213, India*

ARTICLE INFO

Handling Editor: Prof. Paul Kirschner

Keywords:
Proximal Policy Optimization
Deep Deterministic Policy Gradient
Reinforcement Learning
Markov Model
Rendezvous and Docking Mission

ABSTRACT

As spacecraft rendezvous and docking missions become increasingly complex, the need for advanced solutions has surged. In recent years, the application of reinforcement learning techniques to tackle spacecraft rendezvous guidance challenges has emerged as a prominent international trend. Vital to ensuring the secure rendezvous and docking of spacecraft is the task of obstacle avoidance. However, traditional reinforcement learning algorithms lack the ability to enforce safety constraints within the exploration space, which presents a formidable obstacle in the design of spacecraft rendezvous guidance strategies. In response to this challenge, a spacecraft rendezvous guidance methodology founded on safety reinforcement learning is proposed. Firstly, a Markov model is crafted for autonomous spacecraft rendezvous in scenarios involving collision avoidance. A reward system, contingent on obstacle warnings and collision avoidance constraints, is introduced to establish a safety reinforcement learning framework for devising spacecraft rendezvous guidance strategies. Secondly, within the framework of safety reinforcement learning, two deep reinforcement learning (DRL) algorithms, Proximal Policy Optimisation (PPO) and Deep Deterministic Policy Gradient (DDPG), are leveraged to generate these guidance strategies. Experimental findings validate the effectiveness of this approach in successfully executing obstacle avoidance and achieving rendezvous with remarkable precision. Furthermore, through an analysis of the performance and generalization capabilities of these two algorithms, the efficacy of the proposed methodology is further underscored. This fusion of advanced space guidance technology with the principles of Ubiquitous Learning marks a significant step forward in the quest for safer and more efficient spacecraft rendezvous and docking operations.

1. Introduction

The aviation and aerospace industry is expanding quickly, which highlights the urgent need for the development of autonomous spacecraft rendezvous technology, which has recently attracted substantial attention due to an increase in applications in numerous space missions. For this mission, a definite control strategy will be trained using an exploration-adaptive deep deterministic policy gradient (DDPG) algorithm. Autonomous Spacecraft Rendezvous (ASR) refers to the automated tracking and approach of a target spacecraft, with specific constraints on the spacecraft's final position, final velocity, final attitude, formation flight, and other associated tasks. The increasing

[*] Corresponding author.
E-mail addresses: tokpsharma@gmail.com (K.P. Sharma), indradeep1989@gmail.com (I. Kumar), pavitar.19476@lpu.co.in (P.P. Singh), rishianbu@gmail.com (K. Anbazhagan), hmbarakati@uqu.edu.sa (H.M. Albarakati), wasimmohammad71@gmail.com (M.W. Bhatt), avloqulov_anvar@tfi.uz (A.A. Ziyadullayevich), artirana11190@gmail.com (A. Rana), s.a.sivasankari@gmail.com (S.S. A).

https://doi.org/10.1016/j.chb.2023.108110
Received 15 September 2023; Received in revised form 28 November 2023; Accepted 21 December 2023
Available online 24 December 2023

complexity of space missions presents problems for the development of computationally efficient guidance techniques. The predominant approach in addressing this issue is centred on the effective management of dynamic models through optimum control techniques, necessitating substantial computational resources (Pirat et al., 2020; Zappulla et al., 2019a). Understanding system dynamics is crucial for closed-loop (automatic) control as well as open-loop (human) control. These models depend on understanding of the process and are either empirically produced from data or based on more fundamental relationships (first principles, physics-based). The research system's qualities of interest can be summed up using a small number of state variables, and those state variables can also be used to forecast how those attributes will change over time. Simultaneously, as FDeveloping a solution with high computing efficiency for this particular situation using conventional control approaches poses significant challenges (Gao et al., 2022). The use of reinforcement learning methods to address the difficulties associated with spacecraft rendezvous guiding has become a well-known global trend in recent years. Obstacle avoidance is a crucial responsibility for making sure that spacecraft safely rendezvous and dock. But the design of spacecraft rendezvous guidance systems has a significant challenge because conventional reinforcement learning algorithms cannot enforce safety limits in the exploration space. In recent years, the application of reinforcement learning technology to address the challenge of spacecraft rendezvous guiding has gained significant attention on an international scale (Lyu et al., 2020; Sun, 2020). Reinforcement Learning (RL) is a widely employed optimisation control technique that has gained significant traction across various domains, including but not limited to autonomous driving (Zappulla et al., 2019b), mobile edge computing (Hu et al., 2019), recommendation systems (Liu et al., 2023), and industrial Internet of Things (Bengtson et al., 2019). Motivated by intricate guidance objectives, several investigations have incorporated reinforcement learning techniques into the domain of autonomous spacecraft rendezvous with the aim of augmenting the guidance capabilities of spacecraft.

Currently, significant progress has been made in the domain of spacecraft rendezvous navigation through the utilisation of reinforcement learning techniques. In their study, Wang et al. introduced a novel approach for autonomous rendezvous guidance utilising the DDPG algorithm (Long et al., 2022). The authors presented a deep guidance method for spacecraft navigation and employed the distributed deep deterministic policy gradient (D4PG) algorithm (Weiss et al., 2015a). The study conducted by Literature (Xie et al., 2020) focused on investigating the issues related to linear multi-pulse rendezvous mission behaviour cloning and reinforcement learning.

The obstacle avoidance challenge during spacecraft rendezvous is often overlooked in prior studies. The academic community has put up the concept of Safe Reinforcement Learning as a potential solution for addressing safety concerns, such as obstacle avoidance. Previous studies often ignore the challenge of avoiding obstacles during spacecraft rendezvous. The academic community has suggested that safe reinforcement learning is a workable solution for safety concerns, such as obstacle avoidance. The objective of safe reinforcement learning is to maximize cumulative return while guaranteeing compliance with certain security constraints. Regarding the security reinforcement learning, two distinct groups were established. In the first category, the ideal criteria which might include constraint criteria, risk-sensitive criteria, worst-case criteria, and others are modified. Changing the cumulative return maximization goal is one strategy for risk mitigation. Another approach involves confining the exploration space by utilising past information or employing a risk-based mechanism to limit the agent's actions. For instance, within the realm of literature, action-based approaches are employed (McCamish et al., 2010a). In the domain of spacecraft rendezvous guidance. Through the application of reinforcement learning techniques, great progress has been made in the field of spacecraft rendezvous navigation at this time. Using Reinforcement Learning (RL) to regulate satellite rendezvous missions in a closed-loop environment. The algorithm explicitly employed was Proximal Policy Optimisation, or PPO. To prevent any potential collision between the tracking spacecraft and the target spacecraft, the researchers created a safety zone cantered around the target spacecraft. The issue of avoiding collisions with potential hazards during the rendezvous process, such as space debris, should be noted, though, as it was not included in this inquiry. The researchers established a protection zone centred on the target spacecraft to effectively avert any potential collision between the tracking spacecraft and the target spacecraft. However, it is worth noting that this investigation did not address the matter of collision avoidance with obstacles, such as space debris, that may arise during the rendezvous procedure.

This paper employs a model-based approach to attain predictability in reinforcement learning (McCamish et al., 2010b) while investigating the autonomous rendezvous steering of spacecraft in its preliminary research. This paper delves deeper into the issue of obstacle avoidance. Obstacles can be categorised into two types: dynamic obstacles and static obstacles. The avoidance of static barriers is the main topic of our research. We describe a spacecraft rendezvous steering method that is grounded in safety reinforcement learning in this setting. Two deep reinforcement learning (DRL) algorithms, Proximal Policy Optimisation (PPO) and Deep Deterministic Policy Gradient (DDPG), are used to construct the guidance strategy.

To begin, formulate a Markov model to address the autonomous rendezvous of spacecraft in a collision avoidance scenario. Spaceships interacting autonomously in order to avoid colliding. Propose a reward system that also includes collision avoidance limitations and obstacle warning. To be more precise, throughout the rendezvous protocol, designate a defined warning zone that a spacecraft entering it indicates it is in a potentially dangerous position. As of right now, a punitive measure is employed, whereby the spaceship receives a negative return for its behaviour. Additionally, propose a reward mechanism that incorporates obstacle warning and collision avoidance constraints. Specifically, establish a fixed warning zone throughout the rendezvous procedure, where the entry of a spacecraft into this zone signifies its presence in a potentially hazardous situation. Currently, a punitive measure is used whereby a negative reward is bestowed upon the spaceship in response to its actions. Small satellites in Low Earth orbit can use their aerodynamic drag to manage deceleration by varying the surface area exposed to air drag, so modifying their trajectory slightly and preventing debris collisions. A space breakdown, particularly a collision, can release a great deal of energy and send the fragments hurtling in all directions. As there is no air to decelerate the shards, they would all quickly vanish from sight and fly apart. Furthermore, within the safety reinforcement learning framework outlined above, which consists of the Markov model and reward mechanism, the guidance strategy is derived using two deep reinforcement learning (DRL) algorithms, namely Proximal Policy Optimisation (PPO) and Deep Deterministic Policy Gradient (DDPG). The conducted study. The findings demonstrate that the used approach successfully mitigates collisions with obstacles and achieves precise junction completion. The efficacy of this approach is substantiated by an examination of the performance benefits and drawbacks, as well as the generalizability, of the two algorithms. Major Contribution: In order to achieve unpredictability in reinforcement learning, this work uses a model-based approach to examine autonomous spacecraft rendezvous steering in its initial research. This essay explores obstacle avoidance in more detail. This article suggests a reward system based on avoiding collisions while considering a variety of variables and the unique design. The paper is organized into 5 sections, initially, Section 1 provides Introduction section; Section 2 covers the background knowledge; Section 3 presents Implementation, Section 4 presents Experimental Result and Analysis, Section 5 present the conclusion section.

2. Background knowledge

2.1. Spacecraft rendezvous and docking

Spacecraft rendezvous and docking, also known as spacecraft rendezvous and docking, is the process by which two spacecrafts meet (rendezvous) at a predetermined position, speed, and time in orbit, and then align and advance closer until they become structurally connected (docking) (Hu & Chi, 2023). The procedure whereby two spacecraft rendezvous (meet) in orbit at a predefined location, velocity, and time, then align and get closer until they are structurally joined (docking). The tracking spacecraft and the target spacecraft are the terms usually used to describe two spacecraft that are engaged in space rendezvous and coupling, respectively. Usually, the surveillance spacecraft is in operation during the rendezvous and docking process. Two spacecraft performing space rendezvous and coupling are typically referred to as the tracking spacecraft and the target spacecraft, respectively. During the rendezvous and docking procedure, the surveillance spacecraft is typically active, and is typically active. By adjusting the position and attitude of the tracking spacecraft relative to the target spacecraft, the rendezvous and coupling of two spacecraft is accomplished in stages. ASR stands for autonomous spaceship rendezvous and describes the automated tracking and approach of a target spacecraft with particular restrictions on the spacecraft's final position, final velocity, final attitude, formation flight, and other related activities. The development of computationally efficient guidance systems is hindered by the growing complexity. The most popular strategy for resolving this problem is on the efficient management of dynamic models through optimal control methods, which calls for a lot of computer power. For a complete rendezvous and docking mission, the tracking spacecraft's flight phase typically consists of the following stages: preparation stage, launch section, long-distance guidance section, short-range autonomous control section, docking section, combined operation section, evacuation section, return and re-entry section, etc. A tracking spacecraft typically goes through the following stages during its flight phase for a full rendezvous and docking mission: preparation, launch, long-distance guidance, short-range autonomous control, docking, combined operation, evacuation, return and re-entry, etc. The section on short-range autonomous control is the main focus of this paper. Two must be carefully considered in the section on short-range autonomous control. The guidance, navigation and control strategies adopted are different from other stages, which best reflect the characteristics of rendezvous and docking technology, and the entire rendezvous and docking mission has the most stringent performance requirements for the control system in this stage. Consequently, the short-distance autonomous control section has always been the focal point of research into rendezvous and mooring guidance, navigation and control theory and method, and engineering design. Researchers from both home and abroad started integrating artificial intelligence into the rendezvous guidance control of spacecraft as a result of the advancement of AI. The ZEM/ZEV (zero-effort-miss and zero-effort velocity) feedback guidance technique is used in (Scorsoglio et al., 2019). In the literature (Gaudet et al., 2018, pp. 813–827; Turkoglu & Sun, 2018), DRL is used to fit some controllers inside the theoretical framework of optimal control, which enhances the real-time and adaptability of the algorithm. The term "autonomous rendezvous technology" refers to a technique that allows two spacecraft to navigate simultaneously and at the same speed (Tatsch et al., 2006, pp. 276–281).

Fig. 1 depicts the short-range autonomous control segment scenario considered in this article. The target spacecraft is circling the Earth in a circular orbit. There are impediments separating the two spaceships. The tracking spacecraft must be located in distinct locations. As depicted by the dashed line in Fig. 1, the rendezvous with the target spacecraft is completed under the assumption of a collision with obstacles in the centre.

The target spacecraft's centre of mass serves as the origin of the coordinate system. The x-axis corresponds to the line drawn from Earth's geographic centre to the spacecraft's mass centre. To investigate the solar system, a variety of space technology is used. They are used to investigate the solar system's planets, moons, asteroids, and comets. These consist of rovers, landers, orbiters, and fly-bys. This study investigated the use of Reinforcement Learning (RL) in the closed-loop control of satellite rendezvous missions in the area of spacecraft rendezvous guidance. The Proximal Policy Optimisation (PPO) algorithm was used specifically. Target spacecraft's orbit is perpendicular to the y-axis. The right-handed coordinate system is finished off by the z-axis. The centre of mass is the origin of the coordinate system used here, and the Earth is

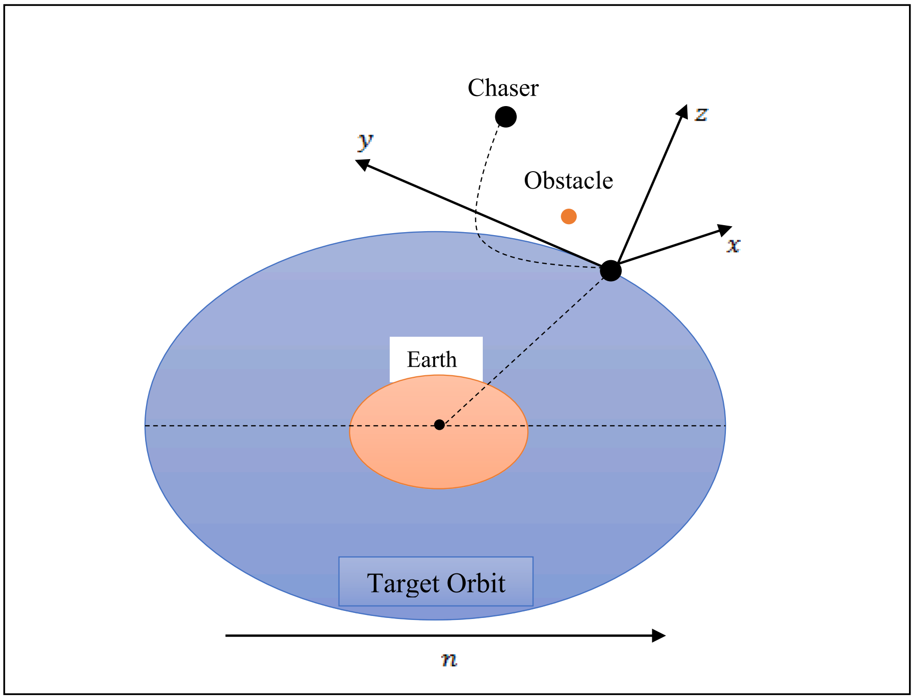


Fig. 1. Spacecraft autonomous rendezvous.

the centre of the circular orbit. The orbital radius, denoted by R, is contrasted by the orbital angular velocity, indicated by n. Newton's theory of motion provides the foundation for the nonlinear relative motion model discussed here (Geng et al., 2021; Mancini et al., 2020).

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 2n\dot{y} + n^2x - \dfrac{\mu(R+x)}{\left[(R+x)^2 + y^2 + z^2\right]^{\frac{3}{2}}} + \dfrac{\mu}{R^2} \\ n^2y - 2n\dot{x} - \dfrac{\mu y}{\left[(R+x)^2 + y^2 + z^2\right]^{\frac{3}{2}}} \\ -\dfrac{\mu z}{\left[(R+x)^2 + y^2 + z^2\right]^{\frac{3}{2}}} \end{bmatrix} + a_f \tag{1}$$

The gravitational constant, denoted as μ, is utilised in the linearization of equation (1) to derive an approximate linear relative motion model, known as the Centre of Gravity Weighted (CGW) equation. This linearization is feasible due to the significantly lesser separation distance between two spacecraft compared to the distance between the target spacecraft and the centre of the Earth.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 3n^2x + 2n\dot{y} \\ -2n\dot{x} \\ -n^2z \end{bmatrix} + a_f \tag{2}$$

The thrust acceleration a_f consists of 3 acceleration values along 3 axes:

$$a_f = \left[\boldsymbol{u}_x, \boldsymbol{u}_y, \boldsymbol{u}_z\right]^{\mathrm{T}} \tag{3}$$

Among them, $\boldsymbol{u_x}, \boldsymbol{u_y}, \boldsymbol{u_z}$ are the control force inputs of the tracking spacecraft in the three axis directions respectively. Then equation (3) can be transformed into the following equations:

$$\begin{cases} \ddot{x} = 3n^2x + 2n\dot{y} + \boldsymbol{u}_x \\ \ddot{y} = -2n\dot{x} + \boldsymbol{u}_y \\ \ddot{z} = -n^2z + \boldsymbol{u}_z \end{cases} \tag{4}$$

The independence of motion in the z direction in relative motion may be observed from equation (4), so allowing for the separate consideration of the controller in the z direction. Calculating one object's motion in relation to another moving object is known as relative motion. As a result, the motion is estimated with respect to the other moving item, rather than the earth, as though the object were in a static condition. It is obvious that the tracking spacecraft will surely collide with things if the incentive system of obstacle warning and collision avoidance constraints is not taken into account. Hence, this study focuses on the problem of spacecraft rendezvous within the confines of a two-dimensional x-y plane. The ultimate equation for relative motion can be expressed as:

$$\begin{cases} \ddot{x} = 3n^2x + 2n\dot{y} + \boldsymbol{u}_x \\ \ddot{y} = -2n\dot{x} + \boldsymbol{u}_y \end{cases} \tag{5}$$

2.2. Reinforcement learning

The field of DRL is a hotspot in the field of reinforcement learning since it combines the benefits of both RL and DL. The inability of conventional reinforcement learning algorithms to enforce safety limits in the exploration space poses a significant challenge to the development of spacecraft rendezvous guidance systems. Reinforcement learning is a data-driven approach in which agent-environment interaction (data) is used to optimize the process. On the other hand, Optimisation Research employs additional techniques that necessitate greater understanding of the issue and/or impose more presumptions. The DRL architecture is depicted in Fig. 2. The agent provides an action plan developed by deep learning that takes into account the present state of affairs. The network takes the current state 's' as input and produces an action an as an output. The environment then upgrades to the next state according to the state transition rules and returns the current reward r according to the reward function. In the subsequent cycle, the agent makes a statement s', which generates an updated action a' that once again influences its surroundings. The agent is constantly gathering information from its surroundings and interacting with it. For the purpose of learning closed-loop control policies, reinforcement learning (RL) is frequently used. Closed-loop control functions well when the sensory feedback it uses is precise, quick, and affordable. This, however, is not always the case.

The DRL algorithm takes these measurements, utilises them to adjust its action plan, and then re-engages with its surroundings to gather more measurements, all in an effort to boost its efficiency. The agent learns the best way to execute the assignment after going through numerous iterations of training.

There are many popular DRL algorithms at present. This article uses the PPO algorithm (Maclean et al., 2014) and the DDPG algorithm (Danielson et al., 2022). The PPO algorithm is an online policy (On-Policy) method, which is a Trust Region Policy Optimisation (TRPO) algorithm. The carried-out study show that the employed method successfully avoids hitting barriers and completes accurate junctions. Examining the performance advantages and disadvantages of the two algorithms, as well as their generalizability, demonstrates the effectiveness of this technique. Compared with the TRPO algorithm, PPO is simpler, easier to implement, and has better performance. The DDPG algorithm is an offline policy (OffGPolicy) method, which is improved on the basis of the DQN algorithm (Silvestre & Ramos, 2023), solving the problem that the original DQN algorithm cannot handle continuous actions and high-dimensional state spaces, and can obtain

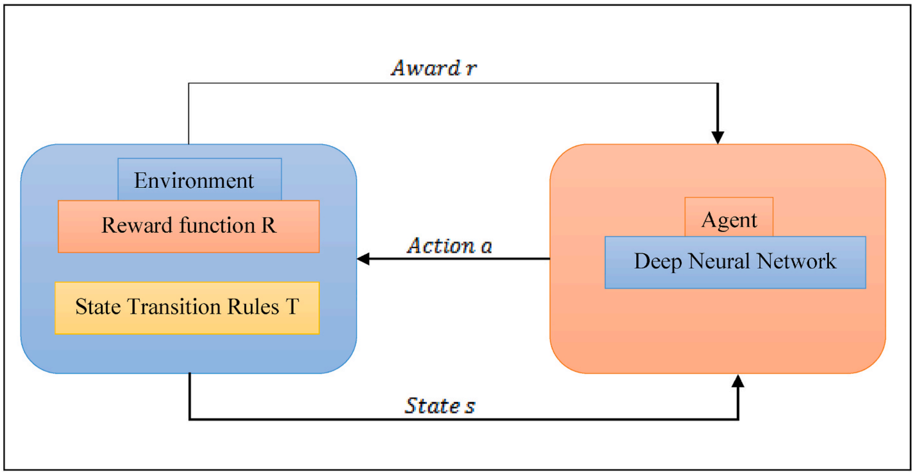


Fig. 2. Deep reinforcement learning frameworks.

deterministic continuous action output based on state information (Li et al., 2021a, 2021b; Weiss et al., 2015b).

3. Implementation

The safety reinforcement learning framework for spacecraft autonomous rendezvous guidance proposed in this article is shown in Fig. 3. The agent outputs the action strategy *a* through the thrust engine according to the current state, and then calculates the next state value *s* according to the state transition equation. The state value is on the one hand It will be input to the agent. On the other hand, it will also be used to calculate the reward value. When calculating the reward value, it is necessary to consider whether the safety constraints are met. Safe reinforcement in cases where it is crucial to guarantee adequate system performance and respect safety requirements during the learning and/or deployment processes, learning can be described as the process of learning policies that maximize the expectation of the return. The final reward is calculated through the reward mechanism based on obstacle warning and collision avoidance constraints proposed in this article. value *r*. Finally, during the training process, it is also necessary to determine whether the current training round meets the termination conditions. The aerospace industry is developing quickly, which highlights the urgent need for autonomous spacecraft rendezvous technology to advance. This technology has recently attracted a lot of attention because it is being used in more space missions. Tracking issue with autonomous spacecraft meeting when collision avoidance is required. To train a specific control strategy for this mission, a deep deterministic policy gradient technique that is exploration-adaptive is developed. This method uses four neural networks, similar to the DDPG algorithm, with two of them being used to construct the deterministic policy and the other two for scoring the acquired policy. On the other hand, adaptive noise is added to lessen the likelihood of oscillations and divergences as well as to eliminate needless computing by reducing the investigation of stabilization difficulties.

Under the above framework, it is necessary to carry out targeted environment setting and algorithm implementation (Zhou et al., 2014a). The rendezvous and coupling of two spacecraft are carried out incrementally by varying the position and attitude of the tracking spacecraft with respect to the target spacecraft. The Markov decision process is created by transforming the rendezvous process. The acceleration command is directly output by the deep reinforcement learning system. Next, we first introduce the environment setting of spacecraft autonomous rendezvous, that is, MDP environment modelling, including scenario assumptions and the design of each element of MDP, and then introduce the algorithm based on PPO and implement the spacecraft rendezvous guidance strategy generation method based on DDPG algorithm (Zhou et al., 2014b, 2014c). Furthermore, Weiss et al. (Weiss et al., 2015c) was able to create spacecraft rendezvous trajectories with a realistic computing load using linear quadratic model predictive control; however, relative motion in this method is restricted to the target spacecraft's orbital plane. Bojarski et al. (Bojarski et al., 2016) mapped camera data to automotive steering commands using a convolutional neural network. Similar to the satellite docking issue, keep-out zones, real-time error handling, and collision avoidance are among the challenging issues involved with autonomous car navigation (Boyarko et al., 2011).

3.1. MDP environment modelling

(1) Scenario Assumptions

As indicated in Section 2.1, the z-directional movement during spacecraft rendezvous is deemed to be independent, hence allowing for the distinct consideration of the z-directional controller. Both spacecraft must be on the same orbital plane and their phases, or relative positions in the orbit, must align in order for an orbital rendezvous to take place. The two vehicles' speeds need to match in order for the docking to occur. The "chaser" is positioned in an orbit that is a little lower than the target's. While docking is the actual physical meeting and joining of two spacecraft, rendezvous is the action of bringing two spacecraft together. We want to maneuver one spacecraft, the Deputy, toward the other, the Chief, so that they can dock. The spaceship rendezvous and docking difficulty is the name of this issue. For resupply and maintenance flights in space applications, it is frequently necessary. This study focuses on the issue of spacecraft rendezvous within the x-y two-dimensional plane, as depicted in Fig. 4. The spacecraft of interest is positioned at the origin, with its orbital angular velocity denoted as n = 0.0011068 rad/s. The designated region for the tracking spacecraft's initial position is represented by a square enclosure, denoted by a dotted box. This enclosure serves as the central point, characterised by coordinates (450 m, 450 m),

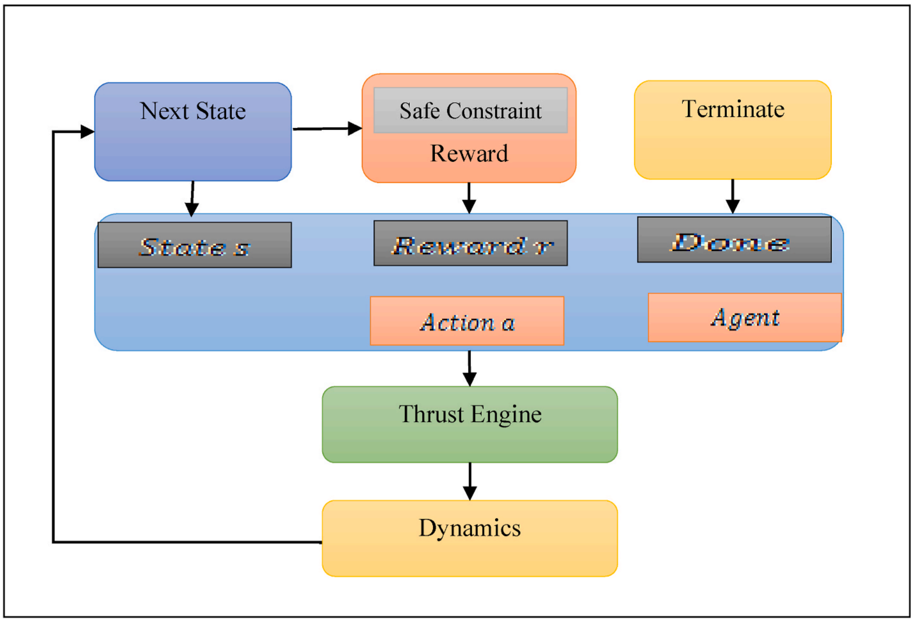


Fig. 3. Safety reinforcement learning framework for Spacecraft Autonomous rendezvous guidance.

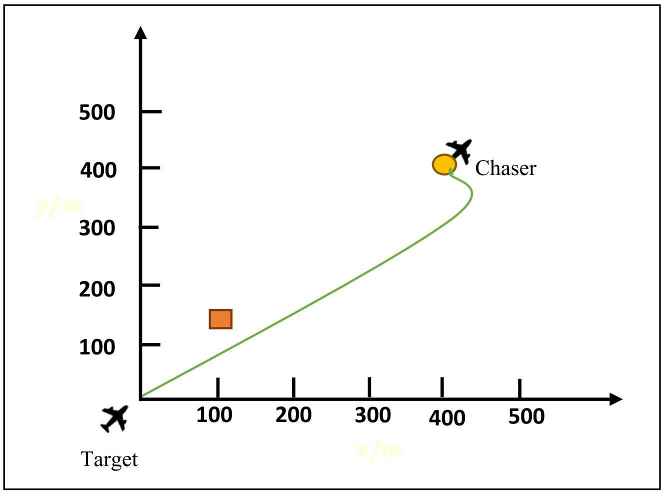


Fig. 4. Two-dimensional plane spacecraft rendezvous scene.

and possesses a side length measuring 100 m. The topic of interest pertains to the measurement of the area of a square. The objective of spacecraft rendezvous is to monitor the spacecraft's initial departure state and manoeuvre the tracking spacecraft to approach the target spacecraft as closely as possible. This is achieved by observing the status information of the tracking spacecraft and adjusting the engine thrust accordingly (Zhou & Li, 2015a). Simultaneously, the process of tracking A designated region comprising of obstructions is situated between the spacecraft and the target vehicle. This region is a square area with a side length of 20 m, centred at the coordinates (110 m, 110 m). It is recommended that the spacecraft refrain from entering this designated region during the rendezvous procedure (Zhou & Li, 2015b).

(2) State Space

The selection of state variables should ensure that it can reflect the rendezvous effect, thereby ensuring that the spacecraft can output correct control instructions according to the observed state (Lucas et al., 2023; Wu et al., 2023; Zhang et al., 2022). The relative position and speed in the CGW equation are key indicators that reflect the approach of the spacecraft, so the selected state variables are: $s = [x, y, \dot{x}, \dot{y}]$, which are the position components and velocity components of the tracking spacecraft along the x-axis and y-axis relative to the target spacecraft respectively. Modifying the optimal criteria falls under the first category. These criteria may include worst-case, risk-sensitive, constraint, and other criteria. Modifying the cumulative return maximization aim is one strategy for reducing risk. Another strategy involves restricting the exploration space using historical data or using a risk-based method to control the agent's behaviour. Such a state setting can ensure that the tracking spacecraft can obtain Sufficient status information about the rendezvous process to ensure the performance of rendezvous control. During the algorithm training process, the position of the tracking spacecraft should be limited to a range to avoid meaningless training if the tracking spacecraft is too far away from the target spacecraft (Frei et al., 2023). In summary, the spatial range of state variables can be set based on the above scenario assumptions, as listed in Table 1.

3) Action Space

The action space consists of two-dimensional vectors, representing the tracking spacecraft engine thrust u_x, u_y in the x-axis direction and y-axis direction respectively. According to the actual hardware conditions, they should be continuous values and should meet the limiting condition, and the range is set to $[-1, 1]$. Here it is assumed that the acceleration of the tracking spacecraft is numerically equal to the thrust of the engine (Niu et al., 2018; Sun et al., 2017; Xia & Zou, 2021).

(4) State Transition Equation

The state transition equation is obtained based on the $C - W$ equation and the Euler method. According to equation (5), the acceleration obtained by the tracking spacecraft in the x-axis and y-axis directions under the current thrust can be calculated (Frei et al., 2023) and the velocity and position of the next state can be updated as follows: (The time step $\Delta t = 1s$ set here):

$$\begin{bmatrix} x' \\ y' \\ \dot{x}' \\ \dot{y}' \end{bmatrix} \leftarrow \begin{bmatrix} x + \dot{x}\Delta t \\ y + \dot{y}\Delta t \\ \dot{x} + \ddot{x}\Delta t \\ \dot{y} + \ddot{y}\Delta t \end{bmatrix} \tag{6}$$

(5) Reward Function

In reinforcement learning, the reward is the main source of information for the agent to judge whether the action is good or bad, so the design of the reward function is very important. This paper proposes a reward mechanism based on collision avoidance, considering many factors, and the specific design is as follows (Aguilar-Marsillach et al., 2023). First, Design the reward according to the termination condition. Define $error = x^2 + y^2 + \dot{x}^2 + \dot{y}^2$, that is, the L2 norm of the state vector to describe the error with the target state, when $error \leq 0.5$, it is considered that the two spacecrafts have completed the rendezvous, the mission is successful, and a larger positive reward will be obtained. However, when the spacecraft times out, collides with obstacles, exceeds the range set in Table 1, etc. during the interaction with the environment, it declares If the mission fails, a large negative reward will be obtained. These rewards are obtained when the tracking spacecraft reaches some key states, so they are sparse rewards. Its composition is listed in Table 2. Among them, steps represent the current cumulative time (Qu et al., 2022; Wang & Wu, 2023). *step* size, and *max _steps* represents the maximum time step size of a round, here it is set to 400s.

Secondly, during the rendezvous process, the state value includes the relative position and velocity of the tracking spacecraft. According to Tables 1 and it can be found that the state space of the tracking spacecraft is very large. If the reward is only given when the termination condition is reached, then the reward will be too sparse and not It is beneficial to complete the task, so it is necessary to design intensive rewards. The intensive reward is the reward that can be obtained at each time step of the tracking spacecraft, which is conducive to the tracking of the spacecraft to complete the task. The specific reward function is listed in Table 3, which is mainly divided into two parts: Error-based rewards and rewards based on obstacle warning and collision avoidance constraints. The description is as follows:

Error-based rewards: In each step, corresponding rewards will be

Table 1
State variable.

State	Scope	Initial Range
x	$[-200m, 600m]$	$400m, 500m$
y	$[-200m, 600m]$	$400m, 500m$
$\dot{x}$	$[-\infty, +\infty]$	$[0\ m/s, 0\ m/s]$
$\dot{y}$	$[-\infty, +\infty]$	$[0\ m/s, 0\ m/s]$

Table 2
Sparse rewards.

Description	Rewards
Successful Rendezvous (error$\leq$0.5)	$+10^* (3 - steps / max\_steps)$
Collision with Obstacles	– 100
Out of bounds	-100
Timeout (steps > max_steps)	-100

Table 3
Dense rewards.

Description	Rewards
Reward based on current state	$-$ 0.001 * (*error*)
If *error* $\leq$ 1.0	+ 1
If entering the warning zone	$-$ 10

given according to the current error value. The smaller the error, the closer the tracking spacecraft is to the target spacecraft, and the greater the reward. The purpose of this design is to make the tracking spacecraft in each step, a reward based on the current state can be obtained, so as to avoid training difficulties caused by too sparse rewards (Sun et al., 2019; Zhang et al., 2020). Especially, when *error* ≤ 1.0, there will be additional rewards to further promote the tracking spacecraft to get closer to the target spacecraft.

Rewards based on obstacle warning and collision avoidance constraints: If a large negative reward is given to the agent only in the event of a collision, even if the final model converges, the tracking spacecraft may still collide with obstacles because the tracking spacecraft cannot use A small amount of collision data is used for effective training. In order to provide early warning during the rendezvous process, a warning zone is set up, that is, an area extending outward within 20 m on each side of the restricted area. In particular, if the tracking spacecraft enters the warning zone, it will not Terminating the current round, instead implementing a soft constraint, the tracking spacecraft will receive a large negative reward at each time step when it enters the warning zone. The purpose of this soft constraint is to help the tracking spacecraft stay away from the restricted area while allowing it to enter the warning zone to collect useful data during training exploration. This approach makes policy updates smoother and can speed up learning and improve the final policy.

3.2. *Realization of PPO-based strategy generation method for rendezvous guidance*

The PPO algorithm is a reinforcement learning algorithm based on policy gradient. It directly learns a policy, that is, $\pi\theta(a|s)$, where θ represents the parameter vector of the policy. The PPO algorithm is a method based on Actor-Critic, that is, there are two Network, one network is used to parameterize the agent's strategy, called Actor; the other network is used to parameterize the value function of the current strategy, called Critic. Based on the current state information, Critic estimates the expected reward, and Actor estimates the strategy, if the reward obtained is better than what the Critic estimated, then the Actor will increase the probability of choosing the action, otherwise it will reduce the probability. The PPO algorithm stabilizes the strategy by setting a single update distance limit, thus improving the Actor G Critic method. By introducing a clipping function,The PPO algorithm optimizes the following objective function:

$$L^{\text{CLIP}}(\boldsymbol{\theta}) = \widehat{E}_t[min(R_t(\boldsymbol{\theta})\widehat{A}_t, \text{clip}(R_t(\boldsymbol{\theta}), 1-\varepsilon, 1+\varepsilon)\widehat{A}_t)] \tag{7}$$

where the ratio $R_t(\boldsymbol{\theta})$ is defined as follows:

$$R_t(\boldsymbol{\theta}) = \frac{\pi_{\boldsymbol{\theta}}(a_t|s_t)}{\pi_{\boldsymbol{\theta}_{\text{old}}}(a_t|s_t)} \tag{8}$$

It describes the difference between the current policy distribution π_θ and the previous policy distribution $\pi_{\theta_{\text{old}}}$. ε is a hyperparameter, usually set to 0.2. $\widehat{A}_t$ is the advantage function, which can be defined in many ways. This article uses the generalized advantage estimation (Generalized Advantage Estimator, GAE) (Geng et al., 2021), defined as follows:

$$\widehat{A}_t^{\text{GAE}(\gamma,\lambda)} = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+1}^V \tag{9}$$

Among them, $\delta_t^V = r_t + \gamma V(s_{t+1}) - V(s_t)$, $V(s_t)$ is the value of state s estimated by Critic, and λ is a hyperparameter. GAE can effectively reduce the variance of gradient estimation and is beneficial to Facing the training of agents when rewards are sparse, this method has been widely used in various DRL algorithm implementations since it was proposed.

The implementation of the PPO-based rendezvous guidance strategy generation algorithm is shown in Algorithm 1.

Algorithm 1. PPO-based rendezvous guidance strategy generation algorithm

```
1. Initialize policy parameters $\theta_0$ and value function parameters $\varphi_0$.
1. 2 .  for $\mathrm{k} = 0, 1, 2, 3\cdots$.
2. 3 .  The tracking spacecraft runs the strategy $\pi_\mathrm{k} = \pi(\theta_\mathrm{k})$ in the rendezvous environment set in Section 3.1, and collects a series of empirical data $\mathscr{D}_\mathrm{k} = \{\tau_\mathrm{i}\}$ according to the state transition equation
3. The cumulative discount reward $\widehat{\mathrm{R}}_\mathrm{t}$ is calculated based on the proposed reward mechanism of obstacle warning and collision avoidance constraints.
4. Calculate the estimated value of the advantage function $\widehat{A}_t$ according to equation (9)
5. Use stochastic gradient ascent to maximize PPO-clip objective function update strategy:
6. $\theta_{k+1} = \text{argmax} \frac{1}{|\mathscr{D}_\mathbf{k}|\mathrm{T}} \sum_{\tau\in\mathscr{Q}_\mathbf{k}} \sum_{\mathrm{t}=0}^{\mathrm{T}} min(\mathrm{R}_\mathrm{t}(\theta)\mathrm{A}^{\pi_{\vartheta_k}}, \text{clip}\ (\mathrm{R}_\mathrm{t}(\theta)$.
7. $1-\varepsilon, 1+\varepsilon)\mathrm{A}^{\pi_{\sigma_k}})$.
8. Update the value function by minimizing the mean square error via stochastic gradient descent:
   $\theta_{k+1} = \text{argmax} \frac{1}{|\mathscr{D}_\mathbf{k}|\mathrm{T}} \sum_{\tau\in\mathscr{Q}_\mathbf{k}} \sum_{\mathrm{t}=0}^{\mathrm{T}} min(\mathrm{R}_\mathrm{t}(\theta)\mathrm{A}^{\pi_{\vartheta_k}}, \text{clip}\ (\mathrm{R}_\mathrm{t}(\theta)$.
9. end for
```

The specific neural network structure and parameters of the PPO algorithm are listed in Table 4 and Table 5 respectively. The input layer of the Actor network is composed of the four-dimensional state space S of the tracking spacecraft. The Actor network outputs the mean and Variance represents the distribution of action values in two directions respectively. The number of nodes in the middle-hidden layer is 256, and the activation function used is *ReLU*. During the training process, the current states is input into the Actor network, and the Actor network outputs actions in both directions. value distribution, and then sample according to these two distributions to obtain specific actions, and use the tanh activation function to limit them between [$-$1,1] to obtain $\boldsymbol{u}_x$ and $\boldsymbol{u}_y$. Track the spacecraft and then interact with the environment to

Table 4
Neural network structure of PPO algorithm.

Layer	Actor Network		Critics Network	
	Number of Nodes	Activation Function	Number of Nodes	Activation Function
Input Layer	4	*ReLU*	4	*ReLU*
Hidden Layer1	256	*ReLU*	256	*ReLU*
Hidden Layer2	256	*ReLU*	256	*ReLU*
Output Layer	2	*Tanh*	1	*Linear*

Table 5
PPO algorithm parameters.

Parameter	Value
Discount factor γ	0.99
Actor Learning Rate	0.00003
Critic Learning Rate	0.00003
Optimizer	*Adam*
Training Rounds	3000
Clip Function Parameter ε	0.2
GAE Hyperparameter λ	0.98
Steps Per Round *steps _per _epoch*	1000
Total Number of *steps total_steps*	3000000

obtain new state, the specific process is: According to the action $\boldsymbol{u}_x$, $\boldsymbol{u}_y$ and $C - W$ equations, that is, equation (5), calculate the acceleration $\ddot{x}$ and $\ddot{y}$ obtained by the tracking spacecraft in the x-axis and y-axis directions respectively under forward thrust, and then according to the state The transfer equation, that is, equation (6), calculates the next state $s^{'} = [x^{'}, y^{'}\dot{x}^{'}, \dot{y}^{'}]$.

The input layer and intermediate hidden layer of the Critic network are consistent with the Actor. The output layer of the Critic network is a one-dimensional scalar used to evaluate the value of the current state, that is $V(s)$. On the one hand, the Critic network optimizes its own parameters through training and evaluates the state. The value makes more accurate predictions. On the other hand, it guides the update direction of the Actor network parameters.

The neural network training optimizer adopts the Adam optimizer, which is the most widely used optimizer in the field of deep learning. In order to do it, the guidance algorithm instructs the spaceship to roll and attack at an angle. The execution of those angle of attack and roll directives is then the responsibility of the control algorithm. Spacecraft uses timely radio signals delivered to and from Earth to navigate. On Earth, navigators monitor its position, speed, and relay course corrections. These methods enable navigators to direct a probe to a precise landing or planetary rendezvous. At the same time, based on the reward function design in Section 3.1, setting the discount factor = 0.99, the PPO algorithm interacts with the environment in total 3 × 106 times.

3.3. Generation method for rendezvous guidance based on DDPG

The DDPG algorithm also adopts the Actor-Critic architecture. The network part consists of the Actor network $\mu(s|\theta^{\mu})$, the critic network $Q(s, a|\theta^{Q})$ and the corresponding actor network $(Target - Actor)\mu(s|\theta^{\mu^{'}}$ and the critic target network (Target-Critic) $Q(s, a|\theta^{Q^{'}})$ corresponding to the critic network, in addition, it also contains random noise N used to increase the ability to explore the environment and provides the network with an offline strategy. Training experience playback pool (Replay-Buffer).

The Actor network uses a set of parameters θ^{μ} to represent the current deterministic strategy, through which the action is output, and the cumulative reward $Q^{\pi} = \mathbb{E}[R_t|s_t, a_t]$ is related to the action, updating θ^{μ} through gradient ascent can make Q^{π} rise. The critic network uses a set of parameters θ^{Q} to estimate the Q value under the current state action. The Q value affects the gradient update of the Actor network in the form of the chain rule. Therefore, the accurate Q value has a very important impact on the network convergence. Through the minimum the θQ is updated by the optimized loss function, which can make the Q value more accurate. The Target-Actor network estimates the target action through the parameter $\theta^{\mu^{'}}$, and the Target-Critic network estimates the target Q value through the parameter $\theta^{Q^{'}}$. Equation (10) gives the parameter update of the target network the method adopts the moving average method, and there is a certain delay with the real network. τ is the moving average coefficient. In practical applications, in order to cut off the data correlation, it is necessary to ensure that there is a certain difference between the target network and the real network, so the value of τ is much smaller $than 1$.

$$\begin{aligned} \theta^{Q^{'}} &\leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q^{'}} \\ \theta^{\mu^{'}} &\leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu^{'}} \end{aligned} \tag{10}$$

The intersection guidance strategy generation algorithm based on DDPG is shown in Algorithm 2.

Algorithm 2. Rendezvous Guidance Strategy Generation Algorithm Based on DDPG

```
1. Initialize the Actor network $\mu(\mathrm{s}|\theta^{\mu})$ and Critic network $\mathrm{Q}(\mathrm{s}, \mathrm{a}|\theta^{\mathrm{Q}})$.
2. Initialize the target networks $\mu^{'}$ and $Q^{'}$, whose weights are consistent with $\mu$ and $Q$ respectively.
3. Initialize experience playback buffer $R$.
4. for $\mathrm{k} = 0, 1, 2, 3\cdots$:.
5. Initialize the stochastic process of tracking the motion exploration of the spacecraft, that is, the noise $\mathscr{N}$.
6. According to 3. Section 1 State space initialization tracking spacecraft state $\mathrm{s}_1$.
7. for $\mathrm{t} = 0, 1, 2, 3\cdots\mathrm{T}$.
8. Select an action according to the current policy and noise $\mathrm{a}_\mathrm{t} = \mu(\mathrm{s}_\mathrm{t}|\theta^{\mu}) + \mathscr{N}\mathrm{t}$.
9. The tracking spacecraft generates corresponding engine thrust according to the action $\mathrm{a}_\mathrm{t}$, and calculates the reward $\mathrm{r}_\mathrm{t}$ and the next state $\mathrm{s}_{\mathrm{t}+1}$ based on the reward mechanism and state transition equation of obstacle warning and collision avoidance constraints.
10. Store empirical data $(\mathrm{s}_\mathrm{t}, \mathrm{a}_\mathrm{t}, \mathrm{r}_\mathrm{t}, \mathrm{s}_{\mathrm{t}+1})$ in $R$.
11. Randomly sample a small batch of transfers $(\mathrm{s}_\mathrm{i}, \mathrm{a}_\mathrm{i}, \mathrm{r}_\mathrm{i}, \mathrm{s}_{\mathrm{i}+1})$ from $R$,
    1. let $\mathrm{r}_\mathrm{i} + \gamma\mathrm{Q}^{'}(\mathrm{s}_{\mathrm{i}+1}, \mu^{'}(\mathrm{s}_{\mathrm{i}+1}|\theta^{\mu^{'}})|\theta^{\mathrm{Q}^{'}})$.
12. Update the critic by minimizing the loss function:
    2. $\mathrm{L} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}}(\mathrm{y}_\mathrm{i} - \mathrm{Q}(\mathrm{s}_\mathrm{i}, \mathrm{a}_\mathrm{i}|\theta^{\mathrm{Q}}))^2$.
13. Update Actor's policy using sampling policy gradient:
    a. $\nabla_{\theta^{''}}\mathrm{J} \approx \frac{1}{\mathrm{N}}\sum_{\mathrm{i}}\nabla_\mathrm{a}\mathrm{Q}(\mathrm{s}, \mathrm{a}|\theta^{\mathrm{Q}})\Big|_{\mathrm{s}=\mathrm{s}_\mathrm{i}, \mathrm{a}=\mu(\mathrm{s}_\mathrm{i})} \nabla_{\theta^{''}}\mu(\mathrm{s}|\theta^{\mu})\Big|_{\mathrm{s}_\mathrm{i}}$.
14. Use equation (10) to update the target network
15. end for
16. end for
```

In order to facilitate subsequent comparative analysis, the neural network architecture of the DDPG algorithm and the PPO algorithm is basically the same, and both algorithms have common parameters (such as discount factor, learning rate, optimizer, etc.) are also consistent. The Actor network input layer and the middle-hidden layer are consistent with the Actor network of the PPO algorithm. The difference is that the output of the Actor network is not a distribution, but a definite action value. The Critic network in addition to the four-dimensional state, the input also adds two-dimensional action variables. The output layer of the Critic network is a one-dimensional scalar, which is used to evaluate the value of the current strategy, that is, $Q(s, a)$, and the rest of the parameters are consistent with the Actor network. The DDPG algorithm has a total of Interacted with the environment 3 × 106 times.

4. Experimental results and analysis

The experiments in this article were all conducted on a Windows 10 workstation. The CPU is a 16-core Intel Xeon E5G2620, the GPU is a GTX 1080Ti, and the workstation memory capacity is 384 GB. Both algorithms were trained for 3 × 106 steps. The programming language used is Python3. 6, and based on PyTorch training network, while using GPU acceleration, the training of PPO algorithm and DDPG algorithm took about 3h and 15h respectively.

4.1. Simulation

Track the state of the spacecraft to perceive itself, including relative position and relative speed, and input it into the policy network, and the policy network outputs the policy. As described in Section 3, the policy network output generated by PPO is two sets of independent Gaussian distributions The mean and variance of, the tracking spacecraft directly uses the mean as a strategy to generate the corresponding engine thrust, while the strategy network generated by the DDPG algorithm outputs deterministic action values, which can be directly used by the tracking spacecraft as a strategy to generate the corresponding engine thrust, and then The next state of the tracking spacecraft is updated according to the state transition equation, and finally the above operations are repeated until the mission is completed.

The experiment sets the initial state $s_0 = [476.13\text{m}, 467.85\text{m}, 0\text{m}/\ \text{s}, 0\text{m}/$, under the control of the trained policy network, the rendezvous trajectory of the spacecraft is shown in Fig. 5. It can be seen that the tracking spacecraft can finally reach the vicinity of the target spacecraft and stabilize there and did not collide with obstacles during the rendezvous process.

Fig. 6(a) and (b) depict the alterations in the relative location and relative velocity of the tracking spacecraft throughout the rendezvous procedure, as influenced by the strategies produced by the PPO algorithm and the DDPG algorithm. The graphic illustrates that as the distance increases, the velocity of the tracking spacecraft also increases. The tracking spacecraft exhibits a proximity to the target spacecraft, resulting in a relative speed that diminishes gradually as the distance between the two entities decreases. Eventually, the relative speed of the tracking spacecraft approaches zero upon nearing the target spacecraft. It is evident that over a period of training, the tracking spacecraft is capable of approaching the target spacecraft and achieving stability in close proximity. Ultimately, the relative location and velocity of the two spacecraft converge towards zero, signifying the successful completion of the rendezvous operation.

To establish the efficacy of the approach described in this article, an additional group of policy networks was trained using the PPO algorithm and DDPG algorithm. Notably, these networks were trained without incorporating the reward mechanism associated with obstacle warning and collision avoidance constraints. Subsequently, a comparative analysis was conducted between the method proposed in this article and the aforementioned alternative approaches. Under the guidance of the expertly educated policy network, The depicted trajectories for spacecraft rendezvous can be observed in Figs. 7 and 8. The trajectory generated by the policy network trained solely by the reward mechanism, without taking into account obstacle warning and collision avoidance restrictions, is shown by the dotted line. On the other hand, the solid line shows the trajectory generated by the policy network trained using the method given in this article. It's evident that the failure to account for the reward system of obstacle warning and collision avoidance limitations will result in the tracking spacecraft inevitably colliding with objects. As per the methodology outlined in this scholarly work, the utilisation of either the PPO or DDPG algorithms can provide the spaceship with the cognitive abilities necessary to evade obstacles and successfully accomplish rendezvous tasks.

In addition, the experiment employed a random selection process to choose 100 test points as the initial positions for spacecraft tracking. This was done to evaluate the efficacy of the strategy created by the strategy network, which was trained using two distinct algorithms, in successfully accomplishing the rendezvous assignment. The outcomes of this evaluation are presented in Table 6. According to the data shown in Tables 6 and it is evident that According to the methodology given in this study, it has been seen that the success rate of both algorithms has achieved a 100% level. However, if we do not take into account the reward mechanism of obstacle warning and collision avoidance requirements, both algorithms exhibit a significant collision rate. This finding serves as evidence for the success of the method suggested in this study.

4.2. Algorithm comparison

In order to compare the performance of the two algorithms and the policy network generated by them, the comparison is made from three perspectives: training efficiency, training time, and task effect.

(1) Training efficiency

The training efficiency is compared from the average round reward, the average round ratio of successful rendezvous, and the average round ratio of collisions.

Fig. 9 shows the learning curves of the two algorithms, which describes the relationship between the average round reward changes during the training process, that is, the average of the cumulative rewards of all rounds, which is also the goal of maximizing the DRL algorithm.

Fig. 10 shows the relationship between the average round ratio of the two algorithms that enable the spacecraft to successfully rendezvous during the training process to measure the training efficiency of the algorithm for this task. The number of iterations required to achieve a higher success rate is fewer, it shows that the training efficiency is higher.

From Fig. 9, the average round reward of the PPO algorithm converges slightly faster, converges at about 2000 iterations, and the growth process is stable, while the DDPG algorithm converges at about 2500 iterations. From Fig. 10, as the training progresses, The DDPG algorithm is the first to reach a 100% success rate, but the DDPG algorithm has a higher floating frequency, and the success rate of PPO algorithm and DDPG algorithm can reach 100% in the end. As space research occurs more frequently, spacecraft run into an increasing number of space obstructions, such as permanent physical structures, other mobile aircraft, and space debris. For orbital flight, avoiding collisions with other space objects is essential. Therefore, robust obstacle avoidance strategies must be put in place to ensure the safe operation of

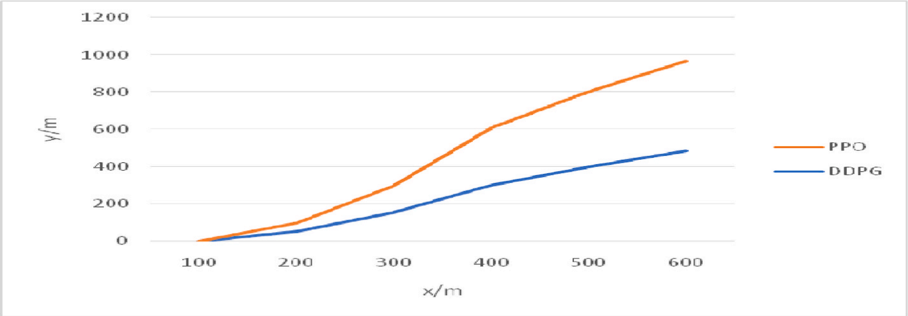
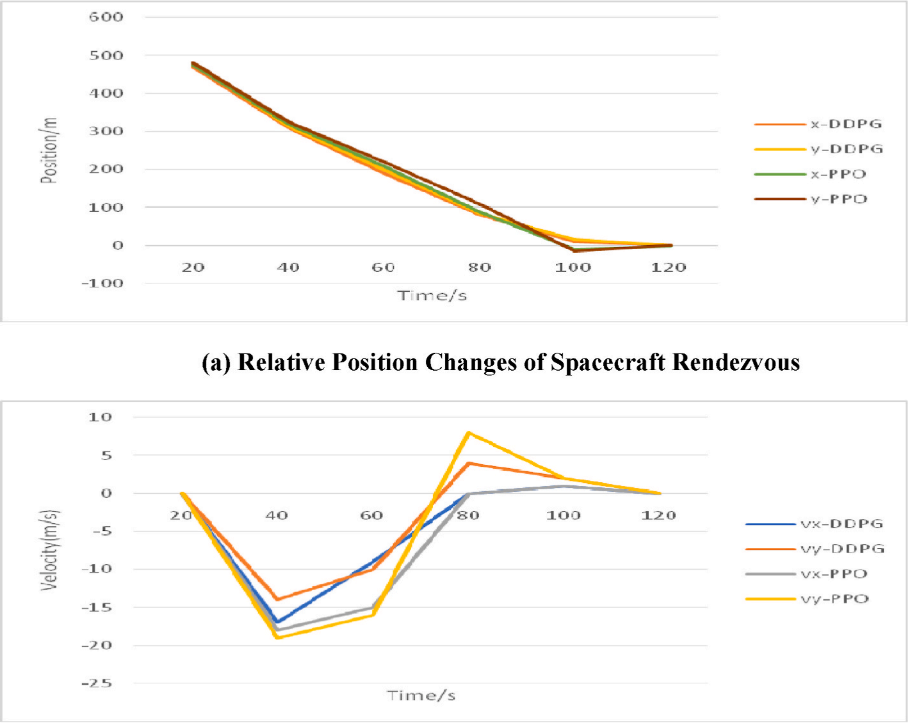


Fig. 5. Spacecraft rendezvous trajectory.



(a) Relative Position Changes of Spacecraft Rendezvous

(b) Relative Velocity Changes of Spacecraft Rendezvous

Fig. 6. (a) relative position changes of spacecraft rendezvous
Fig. 6(b) relative velocity changes of spacecraft rendezvous.

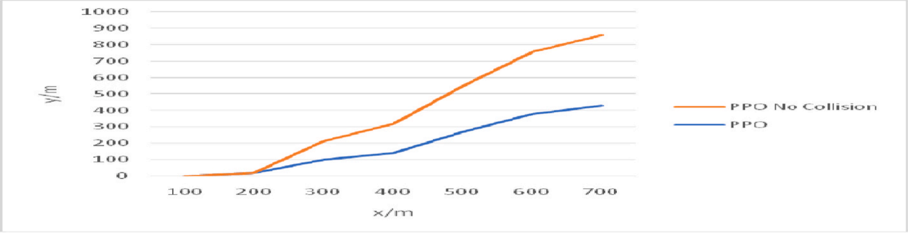


Fig. 7. Intersection trajectory based on PPO algorithm.

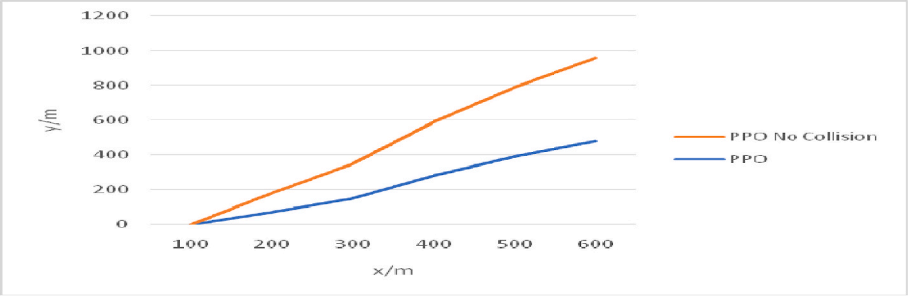


Fig. 8. Intersection trajectory based on DDPG algorithm.

spacecraft. From Fig. 10, the DDPG algorithm will frequently collide with obstacles in the early training. However, the collision probability of the PPO algorithm in the early training is much smaller than that of the DDPG algorithm, and both maintain zero collision probability in the later stage. In summary, the PPO algorithm has higher training efficiency and better stability during the training period.

(2) Training time

In the same environment, based on PyTorch training network, and using GPU acceleration, the DDPG algorithm takes about 15 h to complete the training, while the PPO algorithm only takes about 3 h, that is, the training speed of the PPO algorithm is about 5 times that of the DDPG algorithm. From the table the parameter settings in Tables 5 and

Table 6
Comparison of the success rates of the two algorithms.

Algorithm	Success Rate/ %	Collision Rate/ %	Collision Avoidance Reward Mechanism
PPO	100	0	*Yes*
DDPG	100	0	*Yes*
PPO	69	31	*No*
DDPG	73	27	*No*

7, it can be found that even though the number of interactions between the PPO algorithm and the environment is the same as that of the DDPG algorithm, it is still much faster than the DDPG algorithm. The main reason is that the DDPG algorithm is an offline strategy method, which requires a large experience playback Pool, each sampled data is stored in the pool, and the data will be taken out multiple times to train the network, and in each round of training, it is necessary to read small batches of data from the experience playback pool multiple times to update the network. Compared with the PPO algorithm, the DDPG algorithm updates the network more frequently. In addition, the PPO algorithm is easier to implement and less sensitive to hyperparameters, while the DDPG implementation is complex, and there are many hyperparameters, and the parameter tuning is more cumbersome.

(3) Task effect

The experiment randomly selected 100 test points as the starting point of the tracking spacecraft to test the actual rendezvous effect of the policy network trained by two different algorithms, and the results are listed in Table 7. In the case of tracking the average state value of the final stop of the spacecraft, it can be used to measure the rendezvous accuracy. It can be seen from Table 7 that the success rate of both of them has reached 100%, and they can both achieve high rendezvous accuracy, but in in terms of average reward and average round length, the DDPG algorithm performs better than the PPO algorithm.

4.3. Extensibility testing

To evaluate the generalisation capability of DRL, the experiment included novel test points to assess the performance of the approach when presented with observation states that had not been previously encountered during training. In a similar vein, the experiment employed a random selection process to designate 100 test sites as the initial locations for spacecraft tracking. However, this approach differed from the previous methodology. It is noteworthy that the departure location of the tracking spacecraft has been expanded from the range of 400 m–500 m, as indicated in Table 1, to a broader range of 600 m–800 m. The observation of mission completion in the new state is conducted for both algorithms. The findings are presented in Table 8. According to the data presented in Tables 8 and it is evident that the networks trained by both methods possess the capability to facilitate obstacle avoidance and successfully accomplish the rendezvous task, even when the test range is expanded. The success rate of the mission is 100%, and it continues to maintain a high level of accuracy in rendezvous. This observation indicates that the approach described in this article possesses certain benefits. The capacity for generalisation. Moreover, while comparing the two algorithms, it was seen that the DDPG algorithm exhibited superior performance in terms of average reward and average round length, hence aligning with the findings of previous studies. The analysis conducted in Section 2 demonstrates consistency in the observed outcomes, indicating that the DDPG algorithm yields superior performance in terms of the real task effect.

In summary, the following conclusions can be drawn.

(1) Simulation results show that the method proposed in this paper can effectively avoid obstacles and complete rendezvous with high accuracy.
(2) The algorithm comparison results show that the two algorithms have their own advantages and disadvantages, the PPO algorithm has higher training efficiency and is more stable and faster, while the DDPG algorithm has better performance in actual tasks.

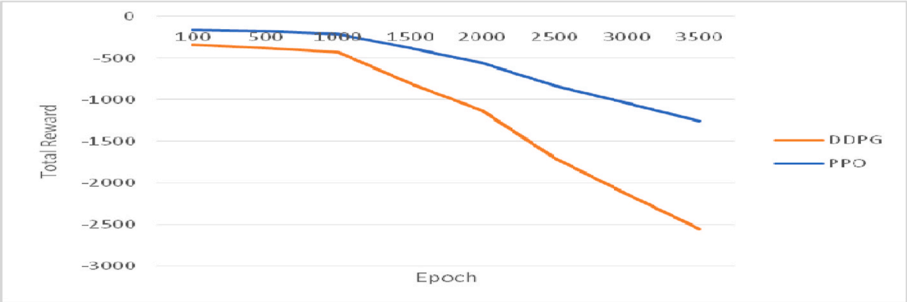


Fig. 9. Average round reward.

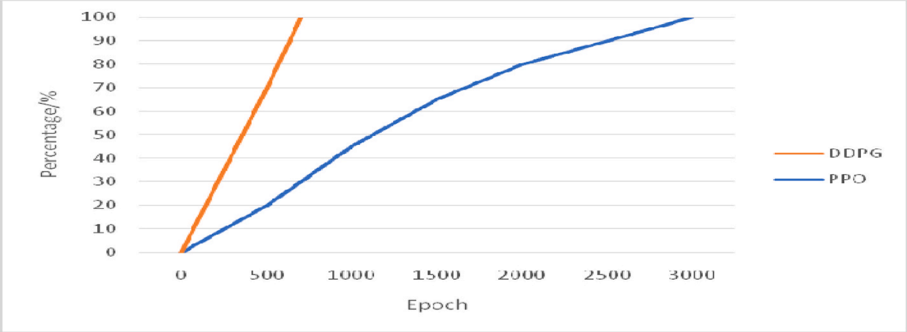


Fig. 10. Average round ratio of successful rendezvous.

Table 7
Comparison of task effects of the two algorithms.

Algorithm	Average Reward	Success Rate/%	Average Intersection Endpoint	Average Round Length
PPO	-137.80	100	$[0.35m, 0.30m, 0.06m/s, 0.03m/s]$	107.56
DDPG	− 121.55	100	$[0.14m, 0.31m, 0.11m/s, 0.08m/s]$	59.40

Table 8
Comparison of the task effects of the two algorithms after expanding the test range.

Algorithm	Average Reward	Success Rate/%	Average Intersection Endpoint	Average Round Length
PPO	-303.30	100	$[0.33m, 0.32m, 0.05m/s, 0.03m/s]$	136.94
DDPG	− 266.20	100	$[0.09m, 0.35m, 0.07m/s, 0.06m/s]$	78.56

(3) The scalability test results show that the method proposed in this article can be effectively applied to tasks that have not been trained and has good generalization ability.

5. Conclusion

The present study introduces a spacecraft rendezvous guidance approach that is grounded in safety reinforcement learning. In this study, we develop a Markov model to address the challenge of autonomous spacecraft rendezvous in collision avoidance scenarios. Additionally, we present a reward mechanism that incorporates obstacle warning and collision avoidance requirements. This approach aims to provide a systematic technique for resolving aerospace issues. This study proposes an academic paradigm for enhancing the safety of vehicle intersection guiding strategies through reinforcement learning techniques. Furthermore, within the context of this safety reinforcement learning framework, a guidance strategy is derived by utilising two DRL algorithms, namely PPO and DDPG. The experimental findings demonstrate that the utilisation of this approach effectively mitigates collisions with obstacles and achieves a greater level of accuracy in accomplishing the rendezvous. Furthermore, a comprehensive evaluation was carried out to compare and assess the scalability of both algorithms, so providing additional evidence to support the efficacy of this approach. In our forthcoming research, we intend to delve deeper into the issue of dynamic obstacle avoidance in spacecraft for the purpose of ensuring safe rendezvous guidance. Specifically, we will explore the use of safety reinforcement learning techniques that incorporate safety shield (Shielding) and runtime monitoring (Runtime Monitor) methodologies. Furthermore, it is imperative to acknowledge that the spaceship guidance and control system represents a safety-critical system, hence necessitating a paramount emphasis on the system's safety and reliability. Consequently, our focus will be on examining formal verification techniques applicable to reinforcement learning models with a specific emphasis on ensuring security.

CRediT authorship contribution statement

Kanta Prasad Sharma: Conceptualization, Data curation, Investigation, Methodology, Validation, Visualization, Writing – original draft. **Indradeep Kumar:** Conceptualization, Data curation, Investigation, Methodology, Resources, Writing – original draft. **Pavitar Parkash Singh:** Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Supervision, Writing – review & editing. **K. Anbazhagan:** Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Writing – original draft. **Hussain Mobarak Albarakati:** Formal analysis, Investigation, Methodology, Resources, Software, Visualization, Writing – original draft. **Mohammed Wasim Bhatt:** Conceptualization, Formal analysis, Investigation, Methodology, Visualization, Writing – original draft. **Avlokulov Anvar Ziyadullayevich:** Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization. **Arti Rana:** Investigation, Resources, Software, Visualization, Writing – review & editing. **Sivasankari S. A:** Investigation, Visualization, Writing – review & editing.

Acknowledgement

The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Aguilar-Marsillach, D., Di Cairano, S., & Weiss, A. (2023). Abort-safe spacecraft rendezvous on elliptic orbits. *IEEE Transactions on Control Systems Technology, 31*(3), 1133–1148. https://doi.org/10.1109/TCST.2022.3216077

Bengtson, M., Hughes, J., & Schaub, H. (2019). Prospects and challenges for touchless sensing of spacecraft electrostatic potential using electrons. *IEEE Transactions on Plasma Science, 47*(8), 3673–3681. https://doi.org/10.1109/TPS.2019.2912057

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., et al. (2016). *End to End Learning for Self-Driving Cars (Version 1)*. arXiv. https://doi.org/10.48550/ARXIV.1604.07316

Boyarko, G., Yakimenko, O., & Romano, M. (2011). Optimal rendezvous trajectories of a controlled spacecraft and a tumbling object. *Journal of Guidance, Control, and Dynamics, 34*(No. 4), 1239–1252.

Danielson, C., Kloeppel, J., & Petersen, C. (2022). Spacecraft attitude control using the invariant-set motion-planner. *IEEE Control Systems Letters, 6*, 1700–1705. https://doi.org/10.1109/LCSYS.2021.3132457

Frei, H., Burri, M., Rems, F., & Eicke-Alexander Risse. (2023). A robust navigation filter fusing delayed measurements from multiple sensors and its application to spacecraft rendezvous. *Advances in Space Research, 72*(Issue 7), 2874–2900. ISSN 0273-1177.

Gao, Y., Li, D., & Ge, S. S. (2022). Time-synchronized tracking control for 6-DOF spacecraft in rendezvous and docking. *IEEE Transactions on Aerospace and Electronic Systems, 58*(3), 1676–1691. https://doi.org/10.1109/TAES.2021.3124865

Gaudet, B., Linares, R., & Furfaro, R. (2018). *Spacecraft rendezvous guidance in cluttered environments via artificial potential functions and reinforcement learning[C]//AAS/AIAA Astrodynamics Specialist Conference*. Univelt Inc., 2018.

Geng, Y., Biggs, J. D., & Li, C. (2021). Pose regulation via the dual unitary group: An application to spacecraft rendezvous. *IEEE Transactions on Aerospace and Electronic Systems, 57*(6), 3734–3748. https://doi.org/10.1109/TAES.2021.3090929

Hu, Q., Chen, W., & Zhang, Y. (2019). Concurrent proximity control of servicing spacecraft with an uncontrolled target. *IEEE, 24*(6), 2815–2826. https://doi.org/10.1109/TMECH.2019.2944387

Hu, Q., & Chi, B. (2023). Spacecraft rendezvous and docking using the explicit reference governor approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 53*(7), 4131–4141. https://doi.org/10.1109/TSMC.2023.3241989

Liu, X., Wang, H., Chen, X., Chen, W., & Xie, Z. (2023). Position awareness network for noncooperative spacecraft pose estimation based on point cloud. *IEEE Transactions on Aerospace and Electronic Systems, 59*(1), 507–518. https://doi.org/10.1109/TAES.2022.3182307

Li, Z., Yu, G., Zhang, Q., Song, S., & Cui, H. (2021a). Adaptive sliding mode control for spacecraft rendezvous with unknown system parameters and input saturation. *IEEE Access, 9*, 67724–67733. https://doi.org/10.1109/ACCESS.2021.3075564

Li, Z., Yu, G., Zhang, Q., Song, S., & Cui, H. (2021b). Adaptive sliding mode control for spacecraft rendezvous with unknown system parameters and input saturation. *IEEE Access, 9*, 67724–67733. https://doi.org/10.1109/ACCESS.2021.3075564

Long, J., Cui, P., & Zhu, S. (2022). Vector trajectory method for obstacle avoidance constrained planetary landing trajectory optimization. *IEEE Transactions on Aerospace and Electronic Systems, 58*(4), 2996–3010. https://doi.org/10.1109/TAES.2022.3143086

Lucas, S., & Zhu, Z. H. (2023). Development of air-bearing microgravity testbed for autonomous spacecraft rendezvous and robotic capture control of a free-floating target. *Acta Astronautica, 203*, 319–328. ISSN 0094-5765.

Lyu, D., Wang, J., He, Z., Hou, B., Zhou, H., & Wang, D. (2020). Navigation and control scheme for space rendezvous and docking with maneuvering noncooperative target based on dynamic compensation. *IEEE Access, 8*, 30174–30186. https://doi.org/10.1109/ACCESS.2020.2972030

Maclean, C., Pagnozzi, D., & Biggs, J. (2014). Planning natural repointing manoeuvres for nano-spacecraft. *IEEE Transactions on Aerospace and Electronic Systems, 50*(3), 2129–2145. https://doi.org/10.1109/TAES.2014.130417

Mancini, M., Bloise, N., Capello, E., & Punta, E. (2020). Sliding mode control techniques and artificial potential field for dynamic collision avoidance in rendezvous maneuvers. *IEEE Control Systems Letters, 4*(2), 313–318. https://doi.org/10.1109/LCSYS.2019.2926053

McCamish, S. B., Romano, M., & Yun, X. (2010a). Autonomous distributed control of simultaneous multiple spacecraft proximity maneuvers. *IEEE Transactions on Automation Science and Engineering, 7*(3), 630–644. https://doi.org/10.1109/TASE.2009.2039010

McCamish, S. B., Romano, M., & Yun, X. (2010b). Autonomous distributed control of simultaneous multiple spacecraft proximity maneuvers. *IEEE Transactions on Automation Science and Engineering, 7*(3), 630–644. https://doi.org/10.1109/TASE.2009.2039010

Niu, J., He, G., Zhang, Y., & Du, X. (2018). Relationship between automation trust and operator performance for the novice and expert in spacecraft rendezvous and docking (RVD). *Applied Ergonomics, 71*, 1–8. ISSN 0003-6870.

Pirat, C., Ankersen, F., Walker, R., & Gass, V. (2020). $\mathcal{H}_{\infty}$ and $\mu $ -Synthesis for Nanosatellites Rendezvous and Docking. *IEEE Transactions on Control Systems Technology, 28*(3), 1050–1057. https://doi.org/10.1109/TCST.2019.2892923

Qu, Q., Liu, K., Wang, W., & Lü, J. (2022). Spacecraft proximity maneuvering and rendezvous with collision avoidance based on reinforcement learning. *IEEE Transactions on Aerospace and Electronic Systems, 58*(6), 5823–5834. https://doi.org/10.1109/TAES.2022.3180271

Scorsoglio, A., Furfaro, R., Linares, R., & Massari, M. (2019). ActorCritic reinforcement learning approach to relative motion guidance in near-rectilinear orbit[C]. In */29th AAS/AIAA space flight mechanics meeting* (pp. 1–20).

Silvestre, D., & Ramos, G. (2023). Model predictive control with collision avoidance for unknown environment. *IEEE Control Systems Letters, 7*, 2821–2826. https://doi.org/10.1109/LCSYS.2023.3288884

Sun, L. (2020). Adaptive Fault-tolerant constrained control of cooperative spacecraft rendezvous and docking. *IEEE Transactions on Industrial Electronics, 67*(4), 3107–3115. https://doi.org/10.1109/TIE.2019.2913826

Sun, L., Huo, W., & Jiao, Z. (2017). Robust nonlinear adaptive relative pose control for cooperative spacecraft during rendezvous and proximity operations. *IEEE Transactions on Control Systems Technology, 25*(5), 1840–1847. https://doi.org/10.1109/TCST.2016.2618907

Sun, Z.-J., Luo, Y.-Z., & Li, H.-Y. (2019). Uncertainty-dependent warning threshold for spacecraft rendezvous collision probability. *IEEE Transactions on Aerospace and Electronic Systems, 55*(1), 2–16. https://doi.org/10.1109/TAES.2018.2845158

Tatsch, A., Fitz-Coy, N., & Gladun, S. (2006). *On-orbit servicing: A brief survey[C]// Proceedings of the 2006 performance metrics for intelligent systems workshop*.

Turkoglu, K., & Sun, F. (2018). Reinforcement learning based continuous-time on-line spacecraft dynamics control: Case study of NASA SPHERES spacecraft[C]//2018. In *AIAA guidance, navigation, and control conference*, 0859.

Wang, M., & Wu, H.-N. (2023). Autonomous game control for spacecraft rendezvous via adaptive perception and interaction. *IEEE Transactions on Aerospace and Electronic Systems, 59*(3), 3188–3200. https://doi.org/10.1109/TAES.2022.3221690

Weiss, A., Baldwin, M., Erwin, R. S., & Kolmanovsky, I. (2015a). Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies. *IEEE Transactions on Control Systems Technology, 23*(4), 1638–1647. https://doi.org/10.1109/TCST.2014.2379639

Weiss, A., Baldwin, M., Erwin, R. S., & Kolmanovsky, I. (2015b). Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies. *IEEE Transactions on Control Systems Technology, 23*(4), 1638–1647. https://doi.org/10.1109/TCST.2014.2379639

Weiss, M., Baldwin, R., Erwin, S., & Kolmanovsky, I. (2015c). Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies. *IEEE Transactions on Control Systems Technology, 23*(No. 4), 1638–1647.

Wu, J., Wei, C., Zhang, H., Liu, Y., Zhang, M., & Wang, H. (2023). Learning-based spacecraft reactive anti-hostile-rendezvous maneuver control in complex space environments. In *Adv. Space Res. ((Vol., 72* pp. 4531–4552). Elsevier BV. https://doi.org/10.1016/j.asr.2023.08.043.

Xia, K., & Zou, Y. (2021). Adaptive saturated fault-tolerant control for spacecraft rendezvous with redundancy thrusters. *IEEE Transactions on Control Systems Technology, 29*(2), 502–513. https://doi.org/10.1109/TCST.2019.2950399

Xie, Y., Zhang, Z., Wu, X., Shi, Z., Chen, Y., Wu, B., & Mantey, K. A. (2020). *Obstacle Avoidance and Path Planning for Multi-Joint Manipulator in a Space Robot, 8* pp. 3511–3526). IEEE Access. https://doi.org/10.1109/access.2019.2961167.

Zappulla, R., Park, H., Virgili-Llop, J., & Romano, M. (2019a). Real-time autonomous spacecraft proximity maneuvers and docking using an adaptive artificial potential field approach. *IEEE Transactions on Control Systems Technology, 27*(6), 2598–2605. https://doi.org/10.1109/TCST.2018.2866963

Zappulla, R., Park, H., Virgili-Llop, J., & Romano, M. (2019b). Real-time autonomous spacecraft proximity maneuvers and docking using an adaptive artificial potential field approach. *IEEE Transactions on Control Systems Technology, 27*(6), 2598–2605. https://doi.org/10.1109/TCST.2018.2866963

Zhang, J., Biggs, J. D., Dong, Y., & Sun, Z. (2020). Finite-time attitude set-point tracking for thrust-vectoring spacecraft rendezvous. *Aerospace Science and Technology, 96*, Article 105588. ISSN 1270-9638.

Zhang, Y., Zhu, B., Cheng, M., & Li, S. (2022). Trajectory optimization for spacecraft autonomous rendezvous and docking with compound state-triggered constraints. *Aerospace Science and Technology, 127*, Article 107733. ISSN 1270-9638.

Zhou, B., & Li, Z.-Y. (2015a). Truncated predictor feedback for periodic linear systems with input delays with applications to the elliptical spacecraft rendezvous. *IEEE Transactions on Control Systems Technology, 23*(6), 2238–2250. https://doi.org/10.1109/TCST.2015.2411228

Zhou, B., & Li, Z.-Y. (2015b). Truncated predictor feedback for periodic linear systems with input delays with applications to the elliptical spacecraft rendezvous. *IEEE Transactions on Control Systems Technology, 23*(6), 2238–2250. https://doi.org/10.1109/TCST.2015.2411228

Zhou, B., Wang, Q., Lin, Z., & Duan, G.-R. (2014a). Gain scheduled control of linear systems subject to actuator saturation with application to spacecraft rendezvous. *IEEE Transactions on Control Systems Technology, 22*(5), 2031–2038. https://doi.org/10.1109/TCST.2013.2296044

Zhou, B., Wang, Q., Lin, Z., & Duan, G.-R. (2014b). Gain scheduled control of linear systems subject to actuator saturation with application to spacecraft rendezvous. *IEEE Transactions on Control Systems Technology, 22*(5), 2031–2038. https://doi.org/10.1109/TCST.2013.2296044

Zhou, B., Wang, Q., Lin, Z., & Duan, G.-R. (2014c). Gain scheduled control of linear systems subject to actuator saturation with application to spacecraft rendezvous. *IEEE Transactions on Control Systems Technology, 22*(5), 2031–2038. https://doi.org/10.1109/TCST.2013.2296044