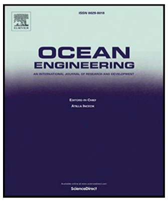
Contents lists available at ScienceDirect

Ocean Engineering

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



Research paper

Target tracking control for Unmanned Surface Vehicles: An end-to-end deep reinforcement learning approach

Zihao Wang [a,b,*], Qiyuan Hu [a], Chao Wang [a,b], Yi Liu [c], Wenbo Xie [a,b]

[a] *Institute of Artificial Intelligence, Shanghai University, Shanghai, 200444, China*
[b] *Engineering Research Center of Unmanned Intelligent Marine Equipment, Ministry of Education, Shanghai, 200444, China*
[c] *Science and Technology on Water Jet Propulsion Laboratory, Marine Design and Research Institute of China, Shanghai, 200011, China*

ARTICLE INFO

Keywords:
Unmanned surface vehicle
Target tracking
Deep reinforcement learning
Target following

ABSTRACT

Target tracking serves as a fundamental motion control function for Unmanned Surface Vehicles (USVs), requiring the USV to rapidly approach a moving target without prior knowledge of its behavior. However, system time lag, underactuation, and environmental disturbances often lead to response delays, degrading tracking efficiency. To address this, we propose a deep reinforcement learning-based end-to-end control method aimed at enhancing the USV's tracking efficiency and responsiveness. Unlike conventional approaches, this method directly learns the mapping from sensor observations to control commands, optimizing decision-making and control actions within a unified framework. A specific deep reinforcement learning algorithm for target tracking is designed based on soft actor–critic framework and integrating predictive target information into the observation space to learn an anticipatory control policy. This paradigm enables the USV to comprehensively account for target movement uncertainty and its own maneuverability under environmental disturbances. Comparative studies are conducted using a high-fidelity simulator that considers the USV's nonlinear dynamics and external influences. The results demonstrate that the proposed method outperforms conventional pure pursuit-based strategy, exhibiting a more efficient and adaptive tracking behavior, akin to human driving habits.

1. Introduction

Unmanned Surface Vehicles (USVs) are a type of marine unmanned system with a wide range of applications, including environmental monitoring, emergency rescue, patrolling and searching, investigation and evidence collection, adversarial gaming, as well as various civilian and military fields. Among their autonomous control capabilities, target tracking is a fundamental function, playing a crucial role in tasks such as rescue, pursuit, and monitoring.

Despite the progress in autonomous guidance and control techniques for USVs, achieving efficient and responsive tracking of maneuvering targets remains a challenge for USVs. These vehicles are inherently charactered by time delays, slow responses to control inputs, underactuated dynamics, and susceptibility to environmental disturbances, all of which significantly impact their ability to track highly maneuverable targets.

The target tracking mission can be divided into two components: the decision-making process, which determines the strategy for approaching the target, and the motion control process, which ensures effective execution of the planned commands. Conventional methods typically address these components separately, with a primary focus on decision-making, commonly referred to as guidance laws.

Classical guidance laws for USV tracking missions are derived from missile interception (Breivik and Fossen, 2008), including the Line of Sight (LOS), Pure Pursuit (PP) and Constant Bearing (CB) methods. These methods adjust the yaw of USVs to achieve a desired heading based on the relative positions of the target and the USV. For example, Breivik et al. (2008) introduced a velocity control system using the CB guidance law to track high-speed linear motion targets, validated through simulations. Similarly, Kim (2020) utilized the CB guidance law for tracking maneuvering targets and proposed a close-range encircling strategy for continuous monitoring. However, unlike high-speed and fast-response missiles, USVs are slower and exhibit time delays in response to control input. This limits their ability to adjust direction as rapidly as missiles do in response to target movements, leading to a behavioral lag when tracking frequently maneuvering targets.

Beyond yaw-guidance-based approaches, several studies have addressed the target tracking problem from the perspective of path planning. For instance, Bibuli et al. (2012) reformulated the problem into

* Corresponding author at: Institute of Artificial Intelligence, Shanghai University, Shanghai, 200444, China.
E-mail address: zihaowang@shu.edu.cn (Z. Wang).

https://doi.org/10.1016/j.oceaneng.2024.120059
Received 12 October 2024; Received in revised form 24 November 2024; Accepted 6 December 2024
Available online 18 December 2024

a virtual target-based path-following task, making the USV follow the past path of the target vessel. Svec et al. (2013) generated tracking trajectories that satisfy the motion constraints of the USV using a hybrid A* approach. Methods leveraging trajectory generation and predictive techniques have also been explored. Švec et al. (2014) applied Monte Carlo sampling to predict target positions and generate feasible trajectories, while Agrawal and Dolan (2015) combined prediction with the A* algorithm to ensure safe paths under maritime collision regulations. These methods mitigate the impact of time lag on USVs by predicting the target's future position, offering improved efficiency. However, they often rely on predefined paths or static planning, which limits adaptability in dynamic and uncertain marine environments.

To efficiently approach a moving target without a predefined trajectory, USV tracking controllers should consider not only the target's current position, but also the anticipated positional changes due to the target's maneuvers. Additionally, it is crucial to comprehensively consider other important aspects, such as kinematic constraints, control system delays, and marine environmental disturbances. This makes the target tracking problem a sequential decision-making problem, where decisions at each step influence the future states and control actions. Moreover, conventional methods often decouple planning and control processes, leading to inconsistencies, as the planned trajectory may not align with the control system's ability to execute it under dynamic and uncertain conditions. These discrepancies can degrade tracking performance, particularly when the system must respond rapidly to frequent target maneuvers. To address this, the controller must dynamically optimize decisions and adjust control inputs in real time, effectively bridging the gap between planning and control while managing uncertainties and disturbances.

Reinforcement learning (RL), particularly deep reinforcement learning (DRL), provides a promising framework for addressing these challenges. By leveraging feedback from the environment, DRL algorithms dynamically optimize decision sequences to achieve long-term objectives. Meanwhile, unlike conventional methods that treat planning and control as separate modules, DRL can integrate these process into a unified framework, enabling end-to-end learning of control policies.

Existing studies have demonstrated the potential of DRL in addressing various USV-related tasks. For instance, multi-agent DRL has been applied to formation planning (Wei et al., 2023), while trust region policy optimization (TRPO) has been used for autonomous berthing under interference conditions (Shimizu et al., 2022). In the context of robust control, Cui et al. (2022) introduced a probabilistic model predictive control framework based on RL to handle environmental disturbances, and Qu et al. (2023) utilized Proximal Policy Optimization (PPO) to train pursuit-evasion strategies in simulated environments. DRL has also been employed in broader USV tasks such as path following (Deraj et al., 2023; Woo et al., 2019; Zhao et al., 2021; Zheng et al., 2022) and collision avoidance (Teitgen et al., 2023; Xu et al., 2023). These studies collectively highlight the potential of DRL to address decision-making and control challenges across a range of USV missions. However, its application to target tracking in USVs, particularly in mitigating the effects of system delays and underactuated dynamics, remains underexplored.

While DRL-based target tracking methods have been explored in other unmanned systems, their extension to USVs poses unique challenges. For example, Zhou et al. (2021) and Bhagat and Sujit (2020) employed DRL to enable UAVs to track ground targets, leveraging the superior maneuverability and minimal response delays of UAVs. Similarly, Sun et al. (2015) proposed a DRL-based tracking approach for autonomous underwater vehicles (AUVs). Although AUVs share certain characteristics with USVs, their fully actuated dynamics and different propulsion mechanisms result in significant differences in maneuverability and control. In contrast, USVs are underactuated systems, and the differences in dynamic characteristics underscore the unique challenges of applying DRL to USVs. To the best of our knowledge, no prior work has specifically investigated DRL methods for USV target tracking, leaving a critical gap in addressing the efficiency and responsiveness required for such missions.

To enhance the target tracking capability of USVs when no prior information about the target's behavior is available, we propose a deep reinforcement learning-based end-to-end control method. This approach integrates predictive target movements with the USV's self-maneuverability, enabling rapid and adaptive pursuit of maneuvering targets while effectively addressing challenges such as system delays, underactuated dynamics, and environmental disturbances. The proposed method leverages the soft actor–critic framework and incorporates predictive target information into the observation space to learn an anticipatory control policy. By unifying decision-making and motion control, our end-to-end system provides an effective solution for the foundational function of target tracking. The method is validated within a high-fidelity simulation environment, demonstrating higher approaching efficiency compared to the pure pursuit method.

The specific contributions of this paper can be summarized as follows

- Proposed a novel DRL-based end-to-end control framework for USV target tracking: The proposed method unifies decision-making and motion control, directly mapping observations to control commands, achieving efficient tracking without prior knowledge of target behavior. This fills a critical gap in the field of real-time USV target tracking, where such an approach has not been previously explored.
- Emphasized the crucial role of predictive information in observations: By incorporating predictive target information into the observation space, the model anticipates future target movements and adapts its control strategy accordingly, improving responsiveness and reducing the effects of temporal delays inherent in USV dynamics.

The rest of the paper is organized as follows. Section 2 presents the formulation of the USV target tracking problem. Then, in Section 3, we present the methodology. Section 4 comprehensively validates the effectiveness of the method through data-driven simulation experiments. Finally, Section 5 summarizes the main findings of this paper and discusses the research contributions.

2. Problem formulation

Given,

(i) a continuous, bounded, non-empty state space $X \subset \mathbb{R}^6$ in which each state $\mathbf{x} = \left[x, y, \varphi, u, v, r\right]^\top$ consists of the position coordinates x and y in the inertial coordinate system, the heading angle φ of the USV model, and the surge velocity u, the sway velocity v in the body-fixed coordinate system, and the yaw rate velocity r, as illustrated in Fig. 1;

(ii) a continuous, bounded and state-dependent control action space $U(\mathbf{x}) \subset \mathbb{R}^2$, where each control action $\mathbf{u}$=$[n, \delta]^\top$ consists of a propeller revolution n and a steering angle δ;

(iii) the current state $\mathbf{x}^t_{\text{usv}} =[x_{\text{usv}}, y_{\text{usv}}, \varphi_{\text{usv}}, u_{\text{usv}}, v_{\text{usv}}, r_{\text{usv}}]^\top_t$ and the control input $\mathbf{u}^t_{\text{usv}} = [n_{\text{usv}}, \delta_{\text{usv}}]^\top_t$ of the USV;

(iv) a 3-degree-of-freedom dynamic model $\dot{\mathbf{x}}_{\text{usv}} = f_{\text{usv}}(\mathbf{x}_{\text{usv}}, \mathbf{u}_{\text{usv}})$ of the USV in the horizontal plane, where $\mathbf{u}_{\text{usv}}$ generates the thrust and torque required for USV motion;

(v) the current state $\mathbf{x}^t_{\text{tar}} =[x_{\text{tar}}, y_{\text{tar}}, \varphi_{\text{tar}}, u_{\text{tar}}, v_{\text{tar}}, r_{\text{tar}}]^\top_t$ of the target vehicle and the control input $\mathbf{u}^t_{\text{tar}} = [n_{\text{tar}}, \delta_{\text{tar}}]^\top_t$ as well as the motion model $\dot{\mathbf{x}}_{\text{tar}} = f_{\text{tar}}(\mathbf{x}_{\text{tar}}, \mathbf{u}_{\text{tar}})$ of the target.

Compute,

a real-time feasible control input $\mathbf{u}^t_{\text{usv}} = [n_{\text{usv}}, \delta_{\text{usv}}]^\top_t$ to bring the USV approach to the target in the shortest possible time.

In this study, it is assumed that the control strategy and dynamic model of the target vehicle are unavailable in a non-cooperative scenario; the movement of the target vehicle is not affected by the tracking

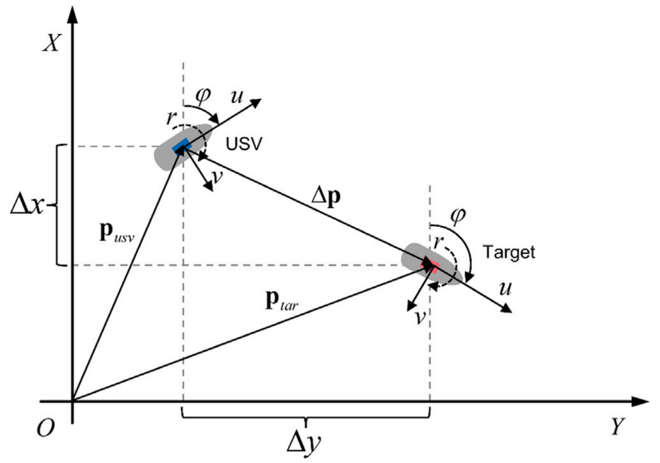


Fig. 1. Coordinate system for USV target tracking scenarios.

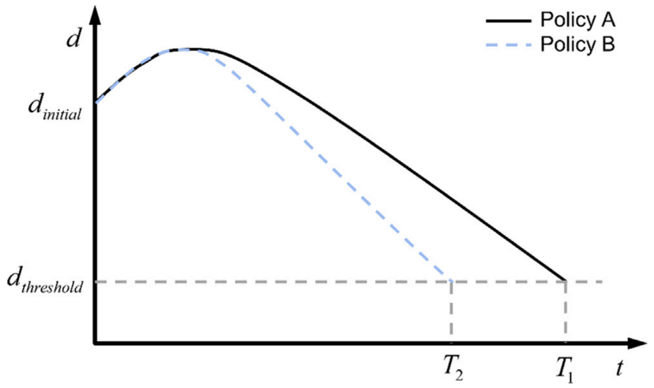


Fig. 2. Relative distance change during target tracking.

actions of the USV. The USV can only perceive the current position of the target vehicle, which makes this problem a Partially Observable Markov Decision Process (POMDP). Additionally, the USV has a higher velocity than the target vehicle, which guarantees the feasibility of the target tracking mission.

During the tracking process, the positions of the USV and the target are denoted as $\mathbf{p}^t_{\rm usv} \triangleq [x_{\rm usv}, y_{\rm usv}]^\top_t \in \mathbb{R}^2$ and $\mathbf{p}^t_{\rm tar} \triangleq [x_{\rm tar}, y_{\rm tar}]^\top_t \in \mathbb{R}^2$, respectively. The relative position relationship between the target vehicle and the USV at time t is defined as follows:

$$\Delta\mathbf{p}^t \triangleq \mathbf{p}^t_{\rm tar} - \mathbf{p}^t_{\rm usv} = [\Delta x, \Delta y]^\top_t. \tag{1}$$

Without loss of generality, the target's position can be obtained through the USV's onboard sensors (e.g., maritime radar). Specifically, the position information is represented by the Euclidean distance $d \in \langle 0, \infty\rangle$ between the target and the USV and the azimuthal angle $\theta \in \langle -\pi, \pi\rangle$ relative to the USV's heading, with the following expressions:

$$d = \sqrt{(\Delta x)^2 + (\Delta y)^2} \tag{2}$$

$$\theta = \arctan 2\,(\Delta y, \Delta x)\,. \tag{3}$$

The target tracking mission aims to accurately and swiftly track a target, minimizing errors and delays in the process. The objective is to control the USV to approach the target vehicle within a predetermined distance threshold $d_{threshold}$. The relative distance change between the USV and the target during the tracking process is illustrated in Fig. 2, where $d_{initial}$ denotes the initial distance between the target and the USV. T_1 and T_2 represent the time required to complete the tracking under two different control policies. In the same scenario, the goal of the tracking task is to reach the predetermined distance $d_{threshold}$ in a shorter time T_2.

The motion control of USVs is characterized by underactuation and temporal hysteresis. Additionally, the movement of USVs is susceptible to environmental disturbances, such as wind, waves, and currents. These challenges significantly impact the ability of USVs to efficiently track targets and impose high requirements on the decision-making capabilities of unmanned systems.

3. Methodologies

3.1. Reinforcement learning algorithm for target tracking

Reinforcement learning provides a feasible approach to solving complex sequential decision-making problems under uncertainty. This is exactly applicable to the target tracking task for USVs, where decisions are made by considering both the target's movements and its own maneuverability under uncertainties.

The intuition behind our approach is to combine the target's current position, predicted future positions, and the USV's state information as observations to design a deep reinforcement learning algorithm that learns the optimal control strategy through continuous experimentation and feedback from the environment. The end-to-end reinforcement learning algorithm directly maps observed states to control signals, integrating conventional planning and control modules. This integration avoids the potential limitations and time lags that arise from designing these modules separately. This reinforcement learning process is similar to how human drivers accumulate experience—by observing and predicting the movement of other vehicles, drivers adjust their actions proactively, thereby achieving more efficient target tracking and the ability to handle complex environments.

Specifically, the observation vector at moment t is defined as:

$$\boldsymbol{s}_t = [d^t, \theta^t, \varphi^t_{\rm usv}, u^t_{usv}, v^t_{\rm usv}, r^t_{\rm usv}, \hat{d}^{t+N}, \hat{\theta}^{t+N}]^\top \in \mathbb{R}^8, \tag{4}$$

where $\hat{d}^{t+N}$ and $\hat{\theta}^{t+N}$ denote the predicted position of the target vehicle N time steps ahead. The prediction of target motion is achieved by fitting a cubic spline curve, as explained in Section 3.2. The end-to-end solution allows the USV to generate a control action $\boldsymbol{a}_t = \delta^t_{\rm usv} \in \langle -30^\circ, 30^\circ\rangle$ by utilizing its own motion information, the current position of the target, and the predicted position of the target.

The target tracking strategy is established based on the Soft Actor–Critic (SAC) algorithm, a model-free reinforcement learning framework based on the Actor–Critic architecture. The SAC algorithm refines the Deep Deterministic Policy Gradient (DDPG) (Lillicrap et al., 2019) by introducing the concept of maximizing entropy, which enhances the model's exploratory capabilities and stability (Haarnoja et al., 2019). An overview of the proposed method is shown in Fig. 3.

To encourage the agent to self-explore and learn effective control policies without much prior knowledge, this paper designs a reward function designed based on the task objective, defined as follows:

$$r_t = r_{dis} + r_{time} = k_1\left(d^t - d^{t+1}\right) - k_2 \tag{5}$$

where r_t denotes the feedback reward that the agent receives from the environment after executing action $\boldsymbol{a}_t$ at time t. r_{dis} represents the distance reward, guiding the USV to approach the target. r_{time} represents the time reward, making the agent aware of the time steps consumed during the tracking process. k_1 and k_2 are hyperparameters used to determine the weights of the distance reward and the time reward, respectively.

The goal of the DRL algorithm is to maximize the return R, which is calculated as follows:

$$R = \sum_{t=0}^{T} r_t = k_1(d^0 - d^{T+1}) - k_2 T, \tag{6}$$

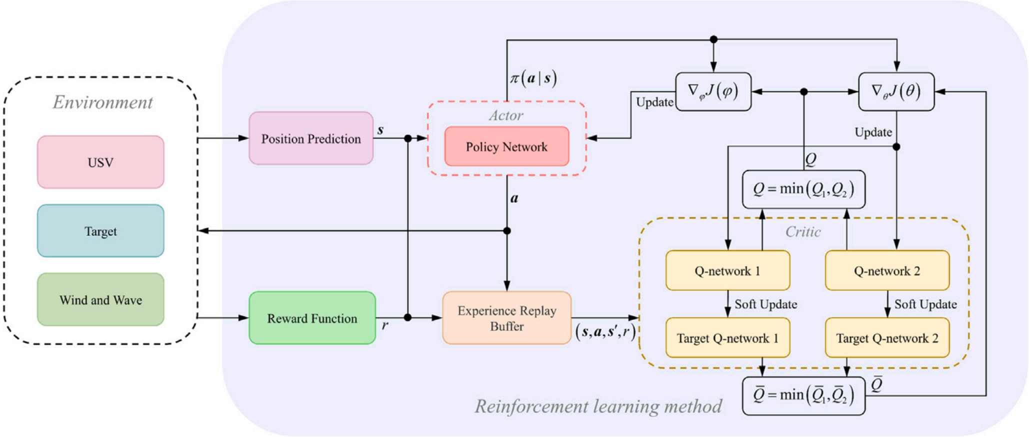


Fig. 3. Overview of proposed methods for USV target tracking.

where T denotes the total time spent during the tracking process. The final cumulative reward for distance is determined by the initial distance d^0 and the final distance d^{T+1} between the USV and the target, as well as the elapsed time T. To prevent unnecessary exploration that could lead to task failure during training, a constant upper limit is imposed on T.

By analyzing the overall outcomes of the tracking task, we can gain a clear understanding of the return. In the early stage of training, the agent is not yet capable of completing the task within the specified time. According to Eq. (6), the return value depends on the initial relative distance d^0 between the USV and the target, as well as the relative distance d^{T+1} at the end state, with k_2T as a constant. Once the agent learns an effective control policy, it can operate within the distance range set by the target navigator within the specified time, making d^{T+1} a constant as well. This return value is then determined by the initial distance d^0 and the task duration T. This reward function helps improve the control strategy from the perspective of tracking efficiency. The designed reward function is highly flexible, as it does not require in-depth knowledge of task specifics. It provides reward signals based directly on agent behavior and the environmental state, reducing rule constraints and minimizing bias introduced by human-defined rules. At the same time, it allows for a broader exploration space, enhancing the agent's decision-making ability and fully utilizing the advantages of deep reinforcement learning (DRL) algorithms.

The optimal policy π^* in SAC is denoted as:

$$\pi^* = \arg\max_{\pi} \sum_t \mathbb{E}_{(s_t,a_t)\sim\rho_\pi} \left[r(s_t,a_t) + \alpha\mathcal{H}(\pi(\cdot|s_t)) \right], \tag{7}$$

where ρ_π represents the state–action marginals of the trajectory distribution. The reward function $r(s_t,a_t)$ reflects the feedback from the environment when the agent takes action a_t in state s_t. Entropy $\mathcal{H}$ measures the stochasticity of policy π under state s_t, and it can be calculated as follows:

$$\mathcal{H}\left(\pi(\cdot \mid s_t)\right) = -\mathbb{E}_{a\sim\pi(\cdot|s_t)} \left[\log \pi(\cdot \mid s_t)\right]. \tag{8}$$

The temperature parameter α determines the weight of entropy compared to reward, thus controlling the explorability of the optimal policy.

To enhance the adaptability of the algorithm, the automatic temperature parameter adjustment is formulated as a constrained optimization problem. The specific form is as follows:

$$\max_{\pi} \mathbb{E}_\pi \left[\sum_t r(s_t,a_t) \right] \quad \text{s.t.} \quad \mathbb{E}_{(s_t,a_t)\sim\rho_\pi}[-\log(\pi_t(a_t|s_t))] \geq \mathcal{H}_0, \tag{9}$$

where $\mathcal{H}_0$ is the desired minimum expected entropy. By maximizing expected returns while ensuring that the mean entropy remains above $\mathcal{H}_0$, the simplified loss function for the temperature parameter is expressed as:

$$J(\alpha) = \mathbb{E}_{a_t\sim\pi_t} \left[-\alpha \log \pi_t(a_t|s_t) - \alpha\mathcal{H}_0\right]. \tag{10}$$

The detailed derivation procedure can be found in the reference (Haarnoja et al., 2019).

During target tracking, the agent utilizes an action value function to evaluate the potential long-term rewards that can be obtained by taking different control actions in a given state. Therefore, the action value function can be expressed as:

$$Q(s_t,a_t) = r(s_t,a_t) + \gamma\mathbb{E}_{s_{t+1}\sim p(s_{t+1}|s_t,a_t)}[V(s_{t+1})], \tag{11}$$

where γ is the discount factor. The state value function is expressed as:

$$V(s_t) = \mathbb{E}_{a_t\sim\pi}[Q(s_t,a_t) - \alpha \log \pi(a_t|s_t)]. \tag{12}$$

In the policy improvement step, for each state, we update the policy according to

$$\pi_{\text{new}} = \arg\min_{\pi'\in\Pi} \mathrm{D_{KL}} \left(\pi'(\cdot|s_t) \middle\| \frac{\exp\left(\frac{1}{\alpha}Q^{\pi_{\text{old}}}(s_t,\cdot)\right)}{Z^{\pi_{\text{old}}}(s_t)} \right), \tag{13}$$

where Π represents the policy set, and $\mathrm{D_{KL}}$ indicates the KL divergence. The logarithm partition function $Z^{\pi_{\text{old}}}(s_t)$ serves to normalize the distribution.

During the training process, two Q-networks with identical structures are employed to mitigate the overestimation of action values. At each iteration, the policy network is updated by selecting the lower Q-value. To enhance sample independence and improve training efficiency, an experience replay buffer D is established to store environment interaction samples (s,a,s',r), which are randomly sampled from D during training. The loss function for the actor networks that train the policy π_φ can be defined as follows:

$$J_\pi(\varphi) = \mathbb{E}_{s_t\sim D, a_t\sim\pi_\varphi} \left[\alpha \log \pi_\varphi\left(a_t \mid s_t\right) - Q_\theta\left(s_t,a_t\right)\right] \tag{14}$$

The loss function for training the critic networks is defined as follows:

$$J_Q(\theta) = \mathbb{E}_{(s_t,a_t,s_{t+1})\sim D, a_{t+1}\sim\pi_\varphi} \left[\frac{1}{2}\left(Q_\theta\left(s_t,a_t\right) - \hat{Q}_\theta\left(s_t,a_t\right)\right)^2 \right], \tag{15}$$

where $\hat{Q}_\theta(s_t,a_t)$ is given by:

$$\hat{Q}_\theta(s_t,a_t) = r(s_t,a_t) + \gamma\left(Q_{\bar{\theta}}(s_{t+1},a_{t+1}) - \alpha \log \pi_\varphi\left(a_{t+1} \mid s_{t+1}\right)\right). \tag{16}$$

The parameters of the target Q-network are denoted as $\bar{\theta}$. The parameters are softly updated in each iteration:

$$\bar{\theta} \leftarrow \tau\theta + (1-\tau)\bar{\theta}, \tag{17}$$

where τ is greater than 0 but much less than 1.

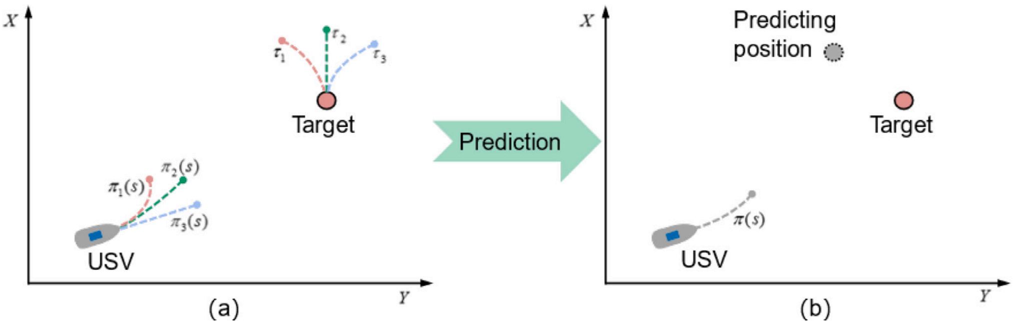


Fig. 4. The influence of forecast information on POMDP modeling.

The detailed algorithm based on the above description is given in Algorithm 1.

Algorithm 1: The DRL algorithm for target tracking control

```
Input: maximum number of tracking time steps $T$, target
       relative distance threshold $d_{threshold}$, the prediction
       horizon $n$.
Initialize parameters $\phi$, $\theta_1, \theta_2$ for the actor network and critic
network and experience replay buffer $\mathcal{D} \leftarrow \emptyset$.
repeat
    Initialize the state of the USV and target vehicle, $\mathbf{x}_{\text{usv}}^{0}$ and
      $\mathbf{x}_{\text{tar}}^{0}$
    while $t \le T$ and the relative distance $\| \Delta\mathbf{p} \| \ge d_{threshold}$ do
        Predict the future position of the target $\hat{\mathbf{p}}_{\text{tar}}^{t+n}$
        Update observation state $\boldsymbol{s}_t \leftarrow \mathbf{x}_{\text{usv}}^{t}, \mathbf{p}_{\text{tar}}^{t}, \hat{\mathbf{p}}_{\text{tar}}^{t+n}$
        Sample the target's action $\mathbf{u}_{\text{tar}}^{t} \sim \rho_{\text{tar}}$ and the USV's
          action $\boldsymbol{a}_t \sim \pi_{\phi}(\boldsymbol{a}_t | \boldsymbol{s}_t)$
        Apply the selected action to either the vehicle dynamics
          model or real-world system
        Generate the next state $\boldsymbol{s}_{t+1} \sim p(\boldsymbol{s}_{t+1} | \boldsymbol{s}_t, \boldsymbol{a}_t)$
        Calculate the reward $r_t$ based on the current state and
          action
        Store the tuple $(\boldsymbol{s}_t, \boldsymbol{a}_t, r(\boldsymbol{s}_t, \boldsymbol{a}_t), \boldsymbol{s}_{t+1})$ in the replay buffer $\mathcal{D}$
    end while
    for each gradient update do
        Update the Q-function parameters
          $\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i)$ for $i \in \{1, 2\}$
        Update policy weights $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$
        Adjust entropy temperature $\alpha \leftarrow \alpha - \lambda \hat{\nabla}_\alpha J(\alpha)$
        Update target network weights
          $\bar{\theta}_i \leftarrow \tau\theta_i + (1 - \tau)\bar{\theta}_i$ for $i \in \{1, 2\}$
    end for
until max episodes;
Output: Optimized parameters $\phi$, $\theta_1, \theta_2$.
```

3.2. Prediction of target position

In the scenario of non-cooperative target tracking, different future trajectories τ_i of the target vehicle correspond to different optimal policies π_i for the USV, making it challenging to learn an effective tracking strategy based solely on the real-time position information of the target, as shown in Fig. 4(a). To address this, we incorporate the predicted information of the target's movement into the observations. This enables the agent to engage in self-exploration considering both the current and future positions of the target. The schematic diagram is illustrated in Fig. 4(b).

Let $\mathbf{p}_{\text{tar}}^{k}$ represent the measured position of the target at time step k, forming a trajectory $[\mathbf{p}_{\text{tar}}^{0}, \mathbf{p}_{\text{tar}}^{1}, \dots, \mathbf{p}_{\text{tar}}^{k}]^\top$. The future position of the target n time steps ahead is represented as $\mathbf{p}_{\text{tar}}^{k+n}$.

The goal is to perform a short-term trajectory forecast of $\hat{\mathbf{p}}_{\text{tar}}^{k+n} = [\hat{x}_{\text{tar}}^{k+n}, \hat{y}_{\text{tar}}^{k+n}]$. A cubic polynomial curve is employed based on the fitting of recent trajectory points. Taking the element $\hat{x}_{\text{tar}}^{k+n}$ as an example, the forecasting mathematical representation is given by:

$$x_{\text{tar}}^{N} = a_0 + a_1 \cdot N + a_2 \cdot N^2 + a_3 \cdot N^3,$$

where $N = k + n$ represents n steps into the future for prediction, with unknown parameters a_0, a_1, a_2, a_3.

To determine these unknown parameters, a matrix representation is utilized:

$$\boldsymbol{X} = \boldsymbol{V} * \boldsymbol{A}, \tag{18}$$

where $\boldsymbol{X} = [x_{\text{tar}}^{k-3}, x_{\text{tar}}^{k-2}, x_{\text{tar}}^{k-1}, x_{\text{tar}}^{k}]^\top \in \mathbb{R}^4$ denotes the target's position at previous and current time steps, $\boldsymbol{A} = [a_0, a_1, a_2, a_3]^\top \in \mathbb{R}^4$ represents the vector of unknown coefficients, and $\boldsymbol{V}$ is the coefficients matrix varying with the time step. The detailed elements are as follows:

$$\begin{bmatrix} x_{\text{tar}}^{k-3} \\ x_{\text{tar}}^{k-2} \\ x_{\text{tar}}^{k-1} \\ x_{\text{tar}}^{k} \end{bmatrix} = \begin{bmatrix} 1 & k-3 & (k-3)^2 & (k-3)^3 \\ 1 & k-2 & (k-2)^2 & (k-2)^3 \\ 1 & k-1 & (k-1)^2 & (k-1)^3 \\ 1 & k & k^2 & k^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{19}$$

The unknown coefficients are solved as $\boldsymbol{A} = \boldsymbol{V}^{-1} * \boldsymbol{X}$. Similarly, we can estimate $\hat{y}_{\text{tar}}^{k+n}$. The relative positions of the target and the USV are calculated as follows:

$$\Delta\hat{\mathbf{p}}^{k+n} \triangleq \hat{\mathbf{p}}_{\text{tar}}^{k+n} - \mathbf{p}_{\text{usv}}^{k} = [\Delta x, \Delta y]_{k+n}^\top. \tag{20}$$

According to Eqs. (2) and (3), $\hat{d}^{k+n}$ and $\hat{\theta}^{k+n}$ can be obtained and incorporated into the observation.

4. Results and discussion

4.1. Experimental setup and training parameter configuration

To evaluate the effectiveness of the proposed method, case studies are conducted in a high-fidelity simulation environment for a 7.5-meter-long USV. The simulator is designed to accurately replicate the dynamic characteristics of USVs and account for environmental disturbances, providing a realistic evaluation platform. It employs a precise maneuvering model that incorporates environmental effects, capturing factors such as system time lag, nonlinear maneuverability, and external disturbances. The dynamical model was developed using a hybrid physical-machine learning approach, which combines mechanistic theory with data-driven techniques based on real navigation data from lake trials. This model has demonstrated strong predictive performance, ensuring that the simulation environment supports the validation of the proposed control method. For more details, refer to Wang et al. (2024). In the case study, the USV has an average speed of approximately 3.6 m/s, while the target vehicle has an average speed of around 3 m/s. Successful tracking is indicated by reducing the relative distance between the target vehicle and the USV to $d_{threshold}$ of

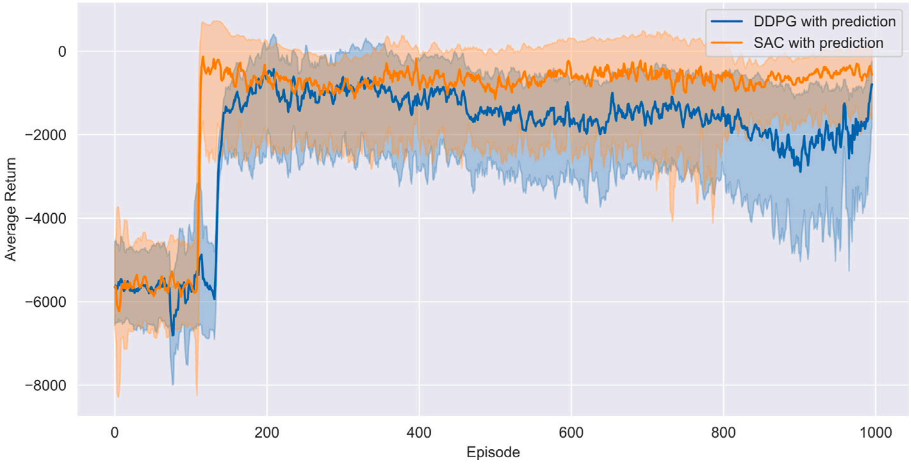


Fig. 5. The return curve for reinforcement learning.

Table 1
Parameters of actor network.

Item	Values
Input layer	8
1st full-connected layer	128
2st full-connected layer	64
Output	1
Learning rate	0.0002
Optimizer	Adam Optimizer

Table 2
Parameters of Value network.

Item	values
Input layer	9
1st full-connected layer	128
2st full-connected layer	64
Output	1
Learning rate	0.002
Optimizer	Adam Optimizer

Table 3
Parameters of training process.

Item	Values
Time step	400
Time interval Δt	0.1s
Prediction time	3s
Replay buffer size	5×10^5
Batch size	128
Discount factor γ	0.99
Learning rate of α	0.0002

$15m$. The performance metrics focus on approaching the target vehicle as quickly as possible.

Tracking is initially performed in a scenario where the target moves in a straight line, using the pure pursuit guidance method as a baseline to validate the tracking efficiency of the proposed method. The DDPG method is also utilized as a control group in this case. Subsequently, we perform scenarios where the target changes course to demonstrate the significance of predictive information in the agent's decision-making. Finally, multiple repetitive experiments are conducted to test the robustness of the method. By tracking vehicles with more complex motion policies, we further illustrate the generalization capabilities of the method.

For the DRL algorithms, the return value of each round is an important indicator of the training results. According to Eq. (6), 10 test evaluations are conducted after each learning round. The relevant hyperparameters for the Actor network, Value network, and parameters of the training process are shown in Tables 1, 2, and 3. The training process incorporates random initialization of the initial states for both the target vehicle and the USV. Additionally, the target vehicle's control strategy is updated through random sampling every 5 s. The return curve after 1000 learning rounds is illustrated in Fig. 5, with a comparison to the DDPG algorithm. In the following figures, the proposed method is labeled as SAC with prediction.

The experimental results show that as learning time increases, the model's performance steadily improves and eventually stabilizes. Compared to the DDPG algorithm, the SAC algorithm not only performs better in terms of average performance but also demonstrates a more stable and decreasing trend in the upper and lower bounds of its returns. This observation validates the core principle of maximum entropy reinforcement learning, which aims to enhance the adaptability and stability of agents in complex and uncertain environments by introducing diversity in their strategies. This stability is crucial for handling complex tasks with uncertainty and volatility in the real world.

4.2. Tracking efficiency for straight-line motion targets

To assess the tracking efficiency of the proposed method, we first conducted experiments using a scenario where the target moves in a straight line. This straightforward motion pattern serves as a fundamental test case, allowing us to validate the basic functionality and performance of our tracking algorithm. For comparison, we employed the pure pursuit guidance method combined with PID control as a benchmark, referred to as the PP method in the following. This strategy aims to align the USV's heading with the target's azimuth angle in real-time. The USV's heading is continuously adjusted through feedback control. The DDPG method is also employed as a control group. This initial test case aims to evaluate the effectiveness of the proposed approach in a controlled environment before exploring more complex motion patterns.

Specifically, the target vehicle moves along a straight line starting from coordinate (0,0) with an initial heading of $\varphi^0_{\mathbf{tar}} = 90°$. The USV's initial coordinate is (−70,0), also with an initial heading of $\varphi^0_{\mathbf{usv}} = 90°$.

The tracking trajectories are shown in Fig. 6. The trajectory controlled by the SAC algorithm approaches the target vehicle more quickly and with less curvature. This indicates that the proposed method identified a more efficient tracking control strategy compared to the other two methods. Fig. 7 illustrates the variation in the relative distance between the USV and the target during the tracking process for the three decision control methods. The results show that by approximately 32 s, the USV controlled by the SAC method reaches

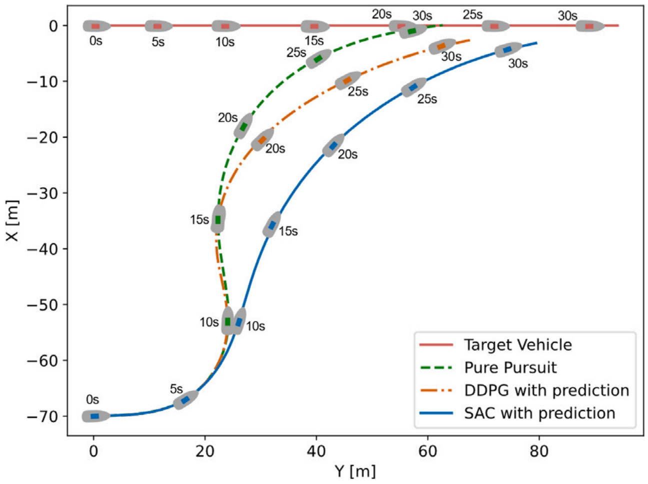


Fig. 6. Tracking trajectory for targets in straight-line motion.

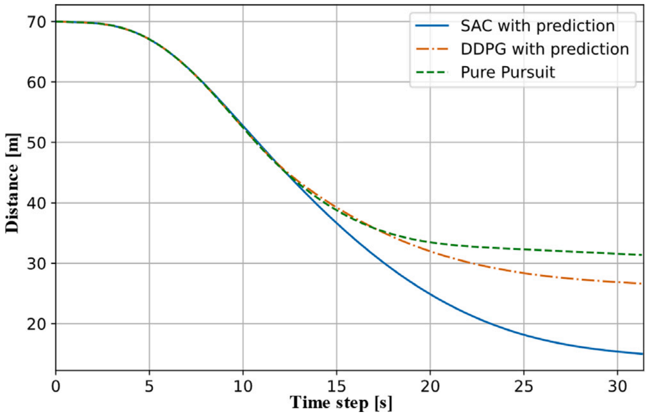


Fig. 7. Relative distance variation during USV target tracking.

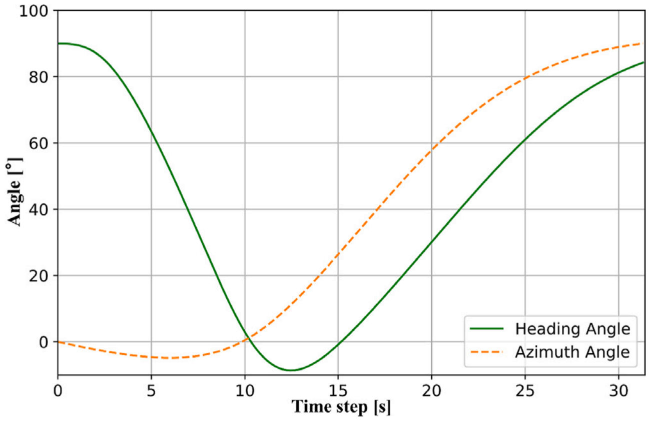


Fig. 8. Heading change controlled by PP.

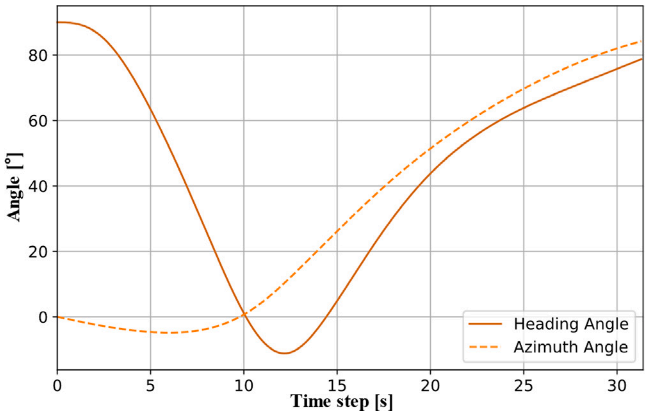


Fig. 9. Heading change controlled by DDPG.

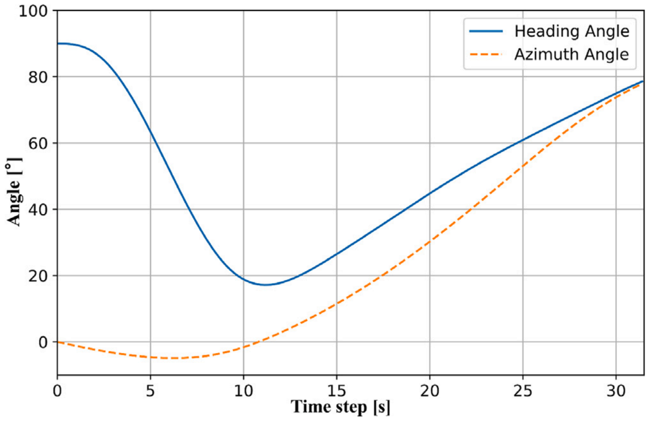


Fig. 10. Heading change controlled by SAC.

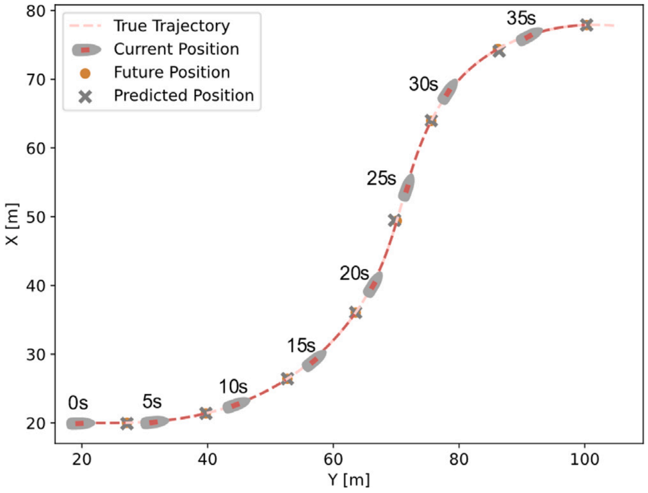


Fig. 11. Predicted results of target vehicle maneuvers.

the target range and successfully completes the tracking, whereas the USV controlled by the PP and DDPG methods still maintains a certain distance. The trajectory generated by the SAC method exhibits less curvature compared to that of the PP and DDPG methods, implying that the SAC method minimizes unnecessary steering actions. In terms of relative distance, the SAC method continuously reduces the distance to the threshold with a steeper trend. This highlights the superior tracking efficiency of the SAC method over the PP and DDPG methods.

Figs. 8, 9 and 10 illustrate the heading angle of the USV and the azimuth angle of the target vessel relative to the USV for three control strategies. Specifically, the azimuth angle corresponds to the desired heading angle for the PP method, as described by Eq. (3). As can be seen, both strategies enable the USV to adjust its heading towards the direction of the target and eventually align with the target's heading. However, the PP method consistently maintains a certain gap from the desired angle. This discrepancy can be attributed to the inherent time

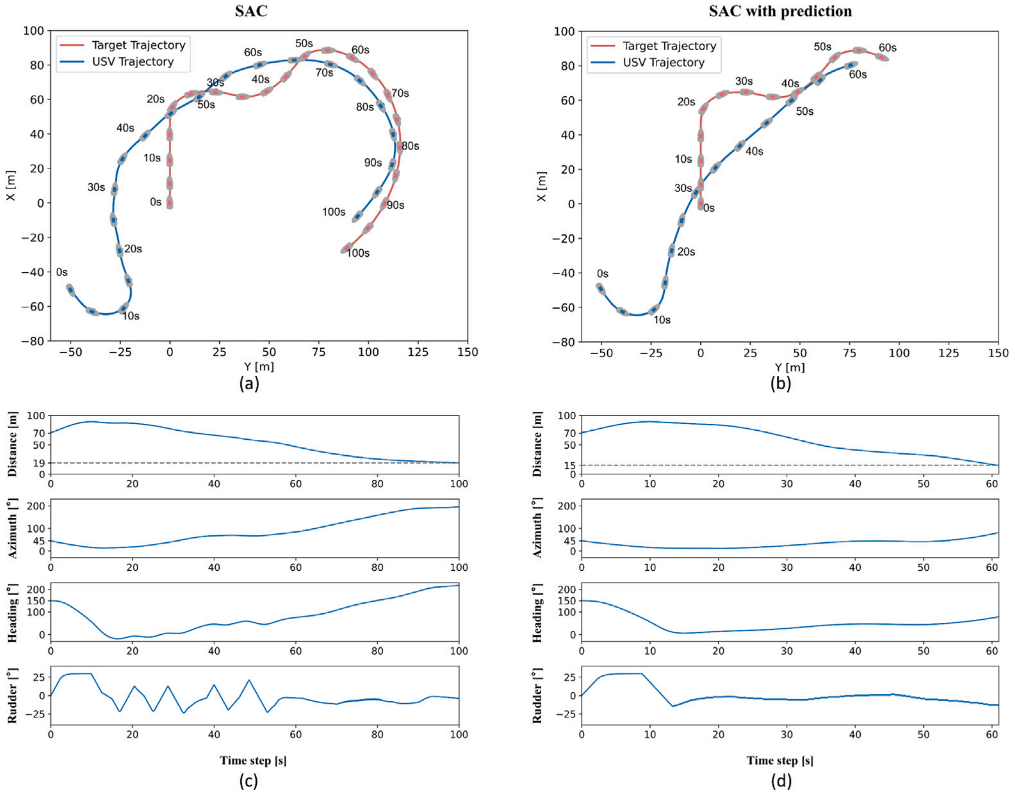


Fig. 12. Tracking for curved motion targets.

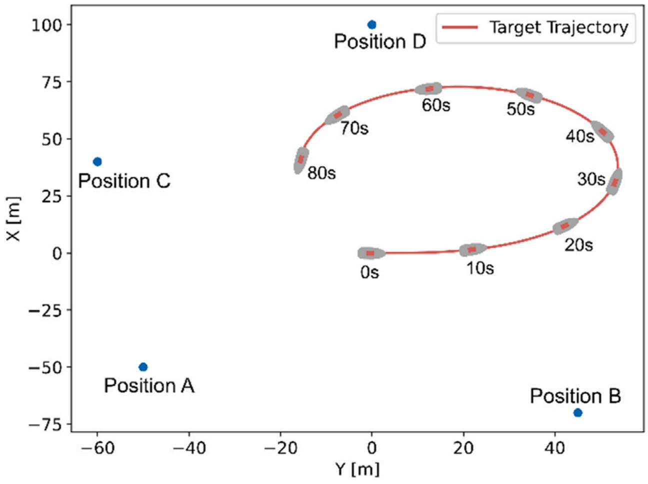


Fig. 13. Target tracking from different initial positions.

lag characteristic of the USV, where the feedback control of the azimuth angle causes the actual heading to lag behind the desired heading. Furthermore, this lag is exacerbated by the continuous changes in the relative direction of the target, which affects the tracking efficiency.

In contrast, two DRL approaches demonstrate a more adaptive and responsive adjustment of the USV's heading angle than the PP method. Notably, the SAC method's heading angle exhibits a look-ahead behavior, pointing toward the target vehicle's future position, thereby improving the tracking efficiency. These results demonstrate that the SAC method reduces the lag effect inherent in the USV's dynamics and outperforms both the PP and DDPG methods in terms of tracking efficiency.

4.3. Tracking performance for curved motion targets

The scenarios of targets moving along curved paths are further tested, where the unavailability of the target's trajectory has a significant impact.

As mentioned in Section 3.2, predictive information is incorporated into the observation to reduce the uncertainty impact caused by the target's maneuvering. The prediction effect is illustrated in Fig. 11. The future position represents the position 3 s after the current moment, while the predicted position denotes the forecasted result. Overall, the outcome is deemed acceptable.

As an example, consider a target with an initial heading $\varphi^0_{\mathbf{tar}}$ of 0° at coordinates (0,0), and the USV starting with an initial heading $\varphi^0_{\mathbf{usv}}$ of 150° at coordinates (−50,−50). The comparison results between the proposed method and the pure SAC without predictive information are depicted in Fig. 12, where each trajectory point interval represents 5 s. The changes in relative distance d, relative azimuth angle θ, heading angle φ, and rudder angle δ during the tracking process are shown sequentially in Fig. 12(c) and Fig. 12(d).

The results demonstrate the advantages of incorporating predictive information in the SAC model for USV target tracking. In the scenario without predictive information (Fig. 12(a)), the SAC model's decisions are based solely on the current state. This results in suboptimal decisions, such as choosing a large left-turning action at 9 s when the target continues straight and then turns right at 20 s. These incorrect decisions lead to frequent course corrections, as indicated by the erratic rudder angle adjustments, and significantly increase the tracking time. The USV takes 100 s to close the distance to 19 m from the target, with pronounced strategy instability due to the limitations of the POMDP model.

Conversely, Fig. 12(b) illustrates the SAC model with predictive information. This model anticipates the target's movements more accurately, reducing strategic uncertainty. Consequently, the USV achieves effective tracking within 62 s, maintaining a distance of 15 m from

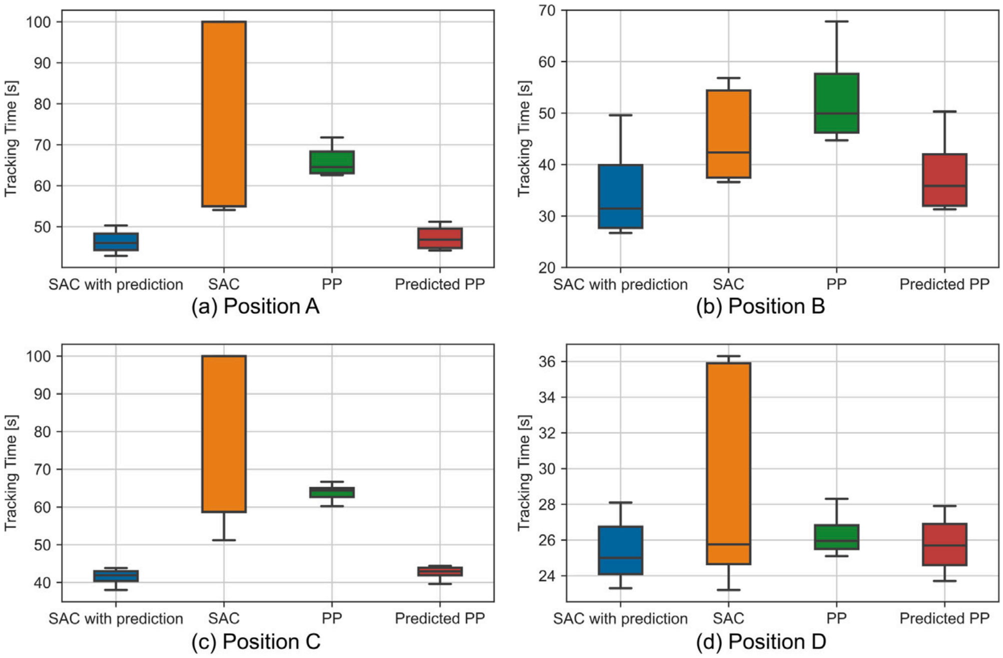


Fig. 14. Tracking time statistics for different initial positions.

the target. This represents a 40% reduction in tracking time, with a smoother trajectory and minimal excessive steering adjustments. These results highlight the enhanced efficiency and accuracy of the SAC model when augmented with predictive information, leading to more stable and rational control decisions.

4.4. Robustness testing in diverse scenarios

To demonstrate the robustness of the method, we conducted simulations to verify its efficacy in diverse scenarios. The test scenario, depicted in Fig. 13, involves the USV tracking the target from four different initial positions labeled “A, B, C, D”. The maneuvering trajectory of the target vehicle is also shown. The USV’s heading is randomly initialized in each simulation to test the method’s robustness. The test is repeated 100 times for each initial position, with the tracking duration recorded for analysis.

We compared the proposed method with the SAC method without predictive information, PP method, and the predicted PP method. The predicted PP method utilizes the predicted position of the target as a reference point and applies the pure pursuit guidance method. This comparison is illustrated in Fig. 13. It is important to note that the fundamental principle of the proposed end-to-end method with predictive information differs from that of the predicted PP method. The proposed method incorporates predictive information directly into the observation vector, which also includes the current position and other relevant data. Through DRL method, the agent comprehensively utilizes this information, learning to make optimal decisions through interaction with the environment.

As seen in Fig. 14, the proposed end-to-end method (SAC with prediction) achieves the smallest median tracking time, with minimal data fluctuation, indicating robust performance. This result highlights the advantage of the proposed end-to-end approach. Compared to the baseline method of predicted PP, which treats decision-making and control separately, the proposed end-to-end method allows the algorithm to continuously learn and adjust in real time, thus avoiding the limitations and delays often associated with separating planning and control into distinct modules. In addition, among the four methods, the methods integrating predictive information effectively reduce the tracking time, thereby alleviating the impact of the USV’s inherent time lag characteristics. The proposed method significantly outperforms the SAC model without prediction, addressing the instability issue inherent in the POMDP model.

While these results demonstrate the effectiveness of the proposed method, reinforcement learning-based control approaches naturally face certain challenges. One aspect is their task-specific design, as components like reward functions are often tailored to particular problems. Extending the approach to more complex tasks may require adjustments to the reward function or incorporating additional decision-making mechanisms. Another aspect is the training cost, particularly in real-world environments where significant resources and time are required. Transfer learning could provide a potential solution by leveraging pre-trained models to reduce training requirements for similar tasks. Despite these challenges, reinforcement learning remains a valuable option for addressing complex decision-making problems under uncertainty, and this study demonstrates its potential in advancing intelligent USV applications.

To further validate the generalizability of the reinforcement learning method, we conducted experiments with targets exhibiting different motion control policies. In these experiments, the initial relative position and headings of the target with respect to the USV were randomly initialized. As shown in Fig. 15, the USV tracked the targets without redundant steering adjustments or significant lags. Even without prior knowledge of the target’s trajectory, the method controls the USV to quickly approach the target to a predetermined distance. This confirms the method’s generalizability and its effectiveness in tracking targets with various control policies.

5. Summary

This paper presents a novel end-to-end control approach for target tracking using reinforcement learning. The method enables USVs to swiftly pursue maneuvering targets without prior knowledge of their trajectories. The designed DRL algorithm can effectively incorporate predictive target information and self-maneuverability into the decision-making process, directly generating control commands in an end-to-end manner. This enables real-time anticipatory control that minimizes tracking delays and improves overall efficiency.

The results underscore the significance of integrating target position prediction information into the observations. Decision-making based

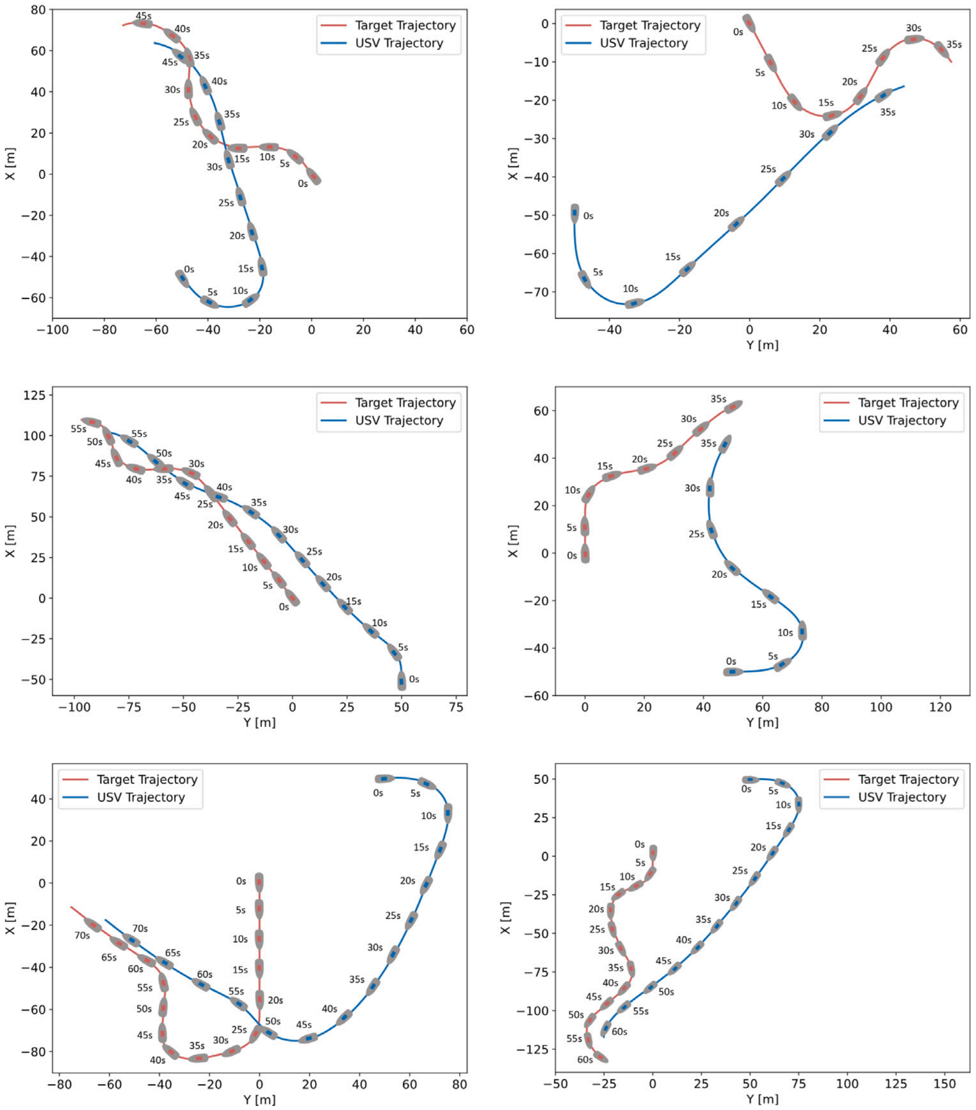


Fig. 15. Tracking trajectories for targets with different motion control policies.

on both the target's current position and its predicted future positions significantly reduces maneuvering uncertainty and enhances the robustness of the DRL method in USV target tracking tasks. The outcome aligns with human driving habits, enabling a predictive response to maneuvering targets. As a fundamental motion control function, this target tracking control framework can be expanded to address more complex decision-making and control problems such as multi-agent pursuit, cooperative encirclement, and monitoring.

CRediT authorship contribution statement

Zihao Wang: Writing – original draft, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Qiyuan Hu:** Writing – original draft, Validation, Software, Methodology, Formal analysis, Conceptualization. **Chao Wang:** Writing – review & editing, Methodology, Formal analysis. **Yi Liu:** Writing – review & editing, Investigation. **Wenbo Xie:** Writing – review & editing, Project administration, Investigation.

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.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 52101361 and No. 62373237).

References

Agrawal, P., Dolan, J.M., 2015. COLREGS-compliant target following for an unmanned surface vehicle in dynamic environments. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, pp. 1065–1070. http://dx.doi.org/10.1109/IROS.2015.7353502.

Bhagat, S., Sujit, P., 2020. UAV target tracking in urban environments using deep reinforcement learning. In: 2020 International Conference on Unmanned Aircraft Systems. ICUAS, pp. 694–701. http://dx.doi.org/10.1109/ICUAS48674.2020.9213856.

Bibuli, M., Caccia, M., Lapierre, L., Bruzzone, G., 2012. Guidance of unmanned surface vehicles: Experiments in vehicle following. IEEE Robot. Autom. Mag. 19, 92–102. http://dx.doi.org/10.1109/MRA.2011.2181784.

Breivik, M., Fossen, T.I., 2008. Guidance laws for planar motion control. In: 2008 47th IEEE Conference on Decision and Control. pp. 570–577. http://dx.doi.org/10.1109/CDC.2008.4739465.

Breivik, M., Hovstein, V.E., Fossen, T.I., 2008. Straight-line target tracking for unmanned surface vehicles. Model. Identif. Control 29, 131–149. http://dx.doi.org/10.4173/mic.2008.4.2.

Cui, Y., Peng, L., Li, H., 2022. Filtered probabilistic model predictive control-based reinforcement learning for unmanned surface vehicles. IEEE Trans. Ind. Inform. 18, 6950–6961. http://dx.doi.org/10.1109/TII.2022.3142323.

Deraj, R., Kumar, R.S., Alam, M.S., Somayajula, A., 2023. Deep reinforcement learning based controller for ship navigation. Ocean Eng. 273, 113937. http://dx.doi.org/10.1016/j.oceaneng.2023.113937.

Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., Levine, S., 2019. Soft actor-critic algorithms and applications. arXiv:1812.05905.

Kim, J., 2020. Target following and close monitoring using an unmanned surface vehicle. IEEE Trans. Syst. Man Cybernet.: Syst. 50, 4233–4242. http://dx.doi.org/10.1109/TSMC.2018.2846602.

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D., 2019. Continuous control with deep reinforcement learning. arXiv:1509.02971.

Qu, X., Gan, W., Song, D., Zhou, L., 2023. Pursuit-evasion game strategy of USV based on deep reinforcement learning in complex multi-obstacle environment. Ocean Eng. 273, 114016. http://dx.doi.org/10.1016/j.oceaneng.2023.114016.

Shimizu, S., Nishihara, K., Miyauchi, Y., Wakita, K., Suyama, R., Maki, A., Shirakawa, S., 2022. Automatic berthing using supervised learning and reinforcement learning. Ocean Eng. 265, 112553. http://dx.doi.org/10.1016/j.oceaneng.2022.112553.

Sun, T., He, B., Nian, R., Yan, T., 2015. Target following for an autonomous underwater vehicle using regularized ELM-based reinforcement learning. In: OCEANS 2015 - MTS/IEEE Washington. pp. 1–5. http://dx.doi.org/10.23919/OCEANS.2015.7401844.

Svec, P., Shah, B.C., Bertaska, I.R., Alvarez, J., Sinisterra, A.J., von Ellenrieder, K., Dhanak, M., Gupta, S.K., 2013. Dynamics-aware target following for an autonomous surface vehicle operating under COLREGs in civilian traffic. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3871–3878. http://dx.doi.org/10.1109/IROS.2013.6696910.

Švec, P., Thakur, A., Raboin, E., Shah, B.C., Gupta, S.K., 2014. Target following with motion prediction for unmanned surface vehicle operating in cluttered environments. Auton. Robots 36, 383–405. http://dx.doi.org/10.1007/s10514-013-9370-z.

Teitgen, R., Monsuez, B., Kukla, R., Pasquier, R., Foinet, G., 2023. Dynamic trajectory planning for ships in dense environment using collision grid with deep reinforcement learning. Ocean Eng. 281, 114807. http://dx.doi.org/10.1016/j.oceaneng.2023.114807.

Wang, Z., Cheng, J., Xu, L., Hao, L., Peng, Y., 2024. Hybrid physics-ML modeling for marine vehicle maneuvering motions in the presence of environmental disturbances. URL https://arxiv.org/abs/2411.13908.

Wei, X., Wang, H., Tang, Y., 2023. Deep hierarchical reinforcement learning based formation planning for multiple unmanned surface vehicles with experimental results. Ocean Eng. 286, 115577. http://dx.doi.org/10.1016/j.oceaneng.2023.115577.

Woo, J., Yu, C., Kim, N., 2019. Deep reinforcement learning-based controller for path following of an unmanned surface vehicle. Ocean Eng. 183, 155–166. http://dx.doi.org/10.1016/j.oceaneng.2019.04.099.

Xu, X., Cai, P., Cao, Y., Chu, Z., Zhu, W., Zhang, W., 2023. Real-time planning and collision avoidance control method based on deep reinforcement learning. Ocean Eng. 281, 115018. http://dx.doi.org/10.1016/j.oceaneng.2023.115018.

Zhao, Y., Qi, X., Ma, Y., Li, Z., Malekian, R., Sotelo, M.A., 2021. Path following optimization for an underactuated USV using smoothly-convergent deep reinforcement learning. IEEE Trans. Intell. Transp. Syst. 22, 6208–6220. http://dx.doi.org/10.1109/TITS.2020.2989352.

Zheng, Y., Tao, J., Sun, Q., Sun, H., Chen, Z., Sun, M., Xie, G., 2022. Soft Actor–Critic based active disturbance rejection path following control for unmanned surface vessel under wind and wave disturbances. Ocean Eng. 247, 110631. http://dx.doi.org/10.1016/j.oceaneng.2022.110631.

Zhou, W., Liu, Z., Li, J., Xu, X., Shen, L., 2021. Multi-target tracking for unmanned aerial vehicle swarms using deep reinforcement learning. Neurocomputing 466, 285–297. http://dx.doi.org/10.1016/j.neucom.2021.09.044.