

# Velocity Domain-based Distributed Pursuit-Encirclement Control for Multi-USVs with Incomplete Information

Yangyang Liu, Chun Liu, *Member, IEEE*, Yizhen Meng, Xiaoqiang Ren, *Member, IEEE*, Xiaofan Wang, *Senior Member, IEEE*

**Abstract**—This paper aims to investigate the distributed pursuit-encirclement control of multiple unmanned surface vehicles (multi-USVs) with incomplete information, characterized by local observation, communication constraint, and velocity absence, in complex environments. A velocity domain-based approach decomposing the pursuit-encirclement issue into velocity planning problem and velocity tracking problem is proposed to deal with this scenario, while considering obstacle avoidance and incomplete information. Firstly, the developed neural network (NN)-based unscented kalman filter (UKF), which is sufficiently real-time and adaptable to environmental features, is established to predict the velocity of the observed USVs. Subsequently, optimal reciprocal collision avoidance (ORCA) is employed to transform the pursuit-encirclement problem into the velocity domain, which enables the planning of optimal velocities that are collision-free while accomplishing the pursuit and encirclement mission. Furthermore, the velocity tracking problem for planned optimal velocity is considered as an optimization problem, where a cost function is designed and solved with adaptive dynamic programming (ADP) technique. Finally, the simulation results verify the effectiveness of the proposed distributed pursuit-encirclement control method.

**Index Terms**—Distributed pursuit-encirclement control, multiple unmanned surface vehicles, incomplete information, NN-based UKF, obstacle avoidance.

## I. INTRODUCTION

MULTIPLE unmanned surface vehicle (USV) systems have found extensive application in maritime collaboration, significantly augmenting the success rate and efficiency of maritime missions [1–3]. In cooperation and coordination missions, pursuit-encirclement is a highly concerned and practical field, especially in military scenarios. However, due to the complexity of real-world environments [4], each mission USV typically has to operate with incomplete information, including communication constraints and local observations [5], [6], and

This work was supported by National Natural Science Foundation of China (62103250, 62273223, and 62333011); Shanghai Sailing Program (21YF1414000); Project of Science and Technology Commission of Shanghai Municipality, China (22JC1401401). (Corresponding author: Chun Liu.)

Y. Y. Liu and X. F. Wang are with the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China (e-mail: yyliu@shu.edu.cn, xfwang@shu.edu.cn).

C. Liu and X. Q. Ren are with the School of Mechatronic Engineering and Automation, and also with the Institute of Artificial Intelligence, Shanghai University, Shanghai 200444, China (e-mail: Chun\_Liu@shu.edu.cn, xqren@shu.edu.cn).

Y. Z. Meng is with the Shanghai Aerospace Control Technology Institute, and also with the Shanghai Key Laboratory of Aerospace Intelligent Control Technology, Shanghai 201109, China (e-mail: 15151830168@163.com).

must avoid collisions with obstacles and other USVs. As a result, efficiently pursuing and encircling an evader through distributed coordination among multi-USVs is a challenging task, particularly in scenarios where the available information is incomplete.

The distributed multi-USV system allows individual USVs to make decisions based on local observations primarily acquired through their onboard sensors [7], [8]. Consequently, it is well-suited for deployment in maritime scenarios with a substantial number of objects, while offering the advantage of reducing computational and hardware costs. One of its main challenges is how to efficiently complete missions in complex marine environments and avoid collisions with other objects in the environment [9], [10]. In [11], a distributed pursuit-encirclement scheme that combines a vision network and a visual perception module is proposed to avoid collisions on the mission. However, its scalability is limited due to its heavy reliance on visual perception, which is not suitable for maritime applications where vehicles on the sea surface may be obscured by waves, fog, or glare. Deep reinforcement learning is employed in [12] to train vehicles for autonomous decision-making and navigation in complex and dynamic environments, enabling them to interact with their surroundings and navigate around obstacles. However, the difference between simulation and reality affects the optimality of the learned strategy. In the above situation, potential field-based methods have been shown to be effective and cost-efficient [13], [14]. Nevertheless, these methods face challenges when dealing with local minima, a common issue in obstacle avoidance methods.

ORCA [15] is a velocity obstacle (VO)-based multi-agent navigation method, which is computationally efficient and won't fall into local minima. The method defines the set of velocities that would result in a collision between two agents within a given time horizon. By choosing velocities outside of this set, agents can avoid collisions with each other. ORCA has earned widespread recognition as a highly effective collision-avoidance method in a variety of scenarios [16–18]. Nevertheless, these existing obstacle avoidance methods require the state of the observed USV to be easily perceived or assume that the velocity of the observed USV is known, which is difficult to achieve in real-world scenarios due to the high cost of precise sensors and complexity of the environment.

Some studies have been conducted on this phenomenon of missing states. A finite-time extended state observer (ESO) is applied to recover the unmeasured linear/angular velocities of

USVs [19]. A simple linear compensation function observer is introduced in [20], exhibiting higher accuracy compared to ESO in state estimations. In [21], a distributed fixed-time estimator is proposed to obtain an estimate of the expected information for each individual in the leader-follower network. Besides, UKF [22] is a nonlinear estimation algorithm that can be used to estimate the states of a nonlinear system, which is particularly useful for practical state estimation applications [23], [24]. However, due to the complexity and variability of the pursuit-encirclement process, conventional state estimation algorithms often suffer from various limitations. These limitations include poor real-time performance, low accuracy, and an inability to adapt to uncertainties, such as those caused by internal modeling errors and external disturbances [25].

Consider the aforementioned challenges, this paper aims to develop a distributed solution for pursuit-encirclement of multi-USVs with incomplete information, including communication loss and limited local observations. Specifically, NNs are utilized to enhance UKF for velocity prediction in the presence of incomplete information. This integration allows the model to adapt to environmental features and makes more accurate predictions. Additionally, ORCA is incorporated into the method to transform the pursuit-encirclement control of multi-USVs into the velocity domain. Finally, a cost function is designed from the perspective of the velocity domain, and ADP is employed to solve the optimal control problem for the mission USV.

The major contributions are summarized as follows:

- 1) An innovative approach is developed to tackle the problem of multi-USV pursuit-encirclement by decomposing it into two sub problems in the velocity domain: velocity planning and velocity tracking of the mission USV. This decomposition utilizes the ORCA avoidance method as an intermediary, offering collision-free guarantee and stable pursuit capabilities, as well as high scalability to accommodate a wide range of pursuit-evasion scenarios.
- 2) An original NN-based UKF velocity prediction method is proposed, combining the characteristics of UKF and NNs. This method allows for precise velocity prediction at any given moment solely based on the observed positions and the previously stored data of the USV. Compared to conventional methods [19], [26], it is capable of adapting to environmental features, offering reliable accuracy and higher real-time performance, providing an innovative solution for USV velocity prediction.
- 3) The proposed method provides a distributed solution for the incomplete information scenario, which is characterized by local observation, communication constraint, and velocity absence. It effectively offers real-time control strategies for the mission USV in complex scenarios involving obstacles and multiple USVs subject to uncertainties.

The remainder of this paper is organized as follows. Preliminaries and problem statement are stated in Section II. In Section III, an NN-based UKF is designed to predict the real-time velocities of the observed USVs, and the optimal velocities are planned for multi-USVs with ORCA method, considering

encirclement and obstacle avoidance. Section IV designs and solves the cost function of the mission USV using ADP to guide the USV in tracking the planned optimal velocities. Section V provides simulations of the pursuit-encirclement control to illustrate the effectiveness of the established method. Finally, conclusions are presented in Section VI.

## II. PRELIMINARIES AND PROBLEM STATEMENT

### A. Optimal reciprocal collision avoidance

As an extension of the VO series algorithm, the ORCA is more suitable for application in obstacle avoidance scenarios with multiple agents and obstacles. For agents  $A$  and  $B$  with radii  $r_A$  and  $r_B$ , respectively, the positions and velocities of each is represented as  $\mathbf{p}_A, \mathbf{p}_B, \mathbf{v}_A, \mathbf{v}_B$ . Let  $D(\mathbf{p}, r) = \{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{p}\| < r\}$  be the set of all points within a distance of radius  $r$  from center  $\mathbf{p}$ , the VO region of agent  $A$  relative to  $B$ , which is shown in Fig. 1, is represented as

$$VO_{A|B}^T = \{\mathbf{v} \mid \exists t \in [0, T] :: t\mathbf{v} \in D(\mathbf{p}_B - \mathbf{p}_A, r_A + r_B)\} \quad (1)$$

It represents the set of all relative velocities of agent  $A$  with respect to  $B$  that would lead to a collision at some moment prior to time  $T$ . Choosing a relative velocity that falls outside the VO region can indeed prevent a collision between agents  $A$  and  $B$ . However, in environments with multiple agents, it is not efficient and far from optimal. In order to make informed decisions, agents should take into account the motion strategies of other agents within their observation range, even in the absence of communication.

ORCA utilizes the VO region to construct a half-plane that represents the set of collision-free velocities for agent  $A$  with respect to agent  $B$ . The ORCA region is shown on the right half of Fig. 1, mathematically denoted as

$$ORCA_{A|B}^T = \left\{ \mathbf{v} \mid \left( \mathbf{v} - (\mathbf{v}_A^{\text{opt}} + \frac{1}{2}\mathbf{u}) \right) \cdot \mathbf{n} \geq 0 \right\} \quad (2)$$

where  $\mathbf{u}$  is the vector pointing from the relative velocity  $\mathbf{v}_A^{\text{opt}} - \mathbf{v}_B^{\text{opt}}$  to the nearest point on the boundary of the VO region  $VO_{A|B}^T$ , with  $\mathbf{v}_A^{\text{opt}}, \mathbf{v}_B^{\text{opt}}$  represented the current velocity of agent  $A$  and  $B$  nominally. The term  $\mathbf{n}$  is the outward normal of the boundary of VO region at point  $(\mathbf{v}_A^{\text{opt}} - \mathbf{v}_B^{\text{opt}}) + \mathbf{u}$ . The vector  $\mathbf{u}$  is defined as

$$\mathbf{u} = \left( \arg \min_{\mathbf{v} \in \partial VO_{A|B}^T} \|\mathbf{v} - (\mathbf{v}_A^{\text{opt}} - \mathbf{v}_B^{\text{opt}})\| \right) - (\mathbf{v}_A^{\text{opt}} - \mathbf{v}_B^{\text{opt}}) \quad (3)$$

The collision-free velocity set  $ORCA_{B|A}^T$  of agent  $B$  is defined symmetrically to agent  $A$ , according to which,  $ORCA_{A|B}^T$  and  $ORCA_{B|A}^T$  are reciprocally collision-avoiding and maximal [15].

### B. Underactuated USV Model

The underactuated USV model, which has three degrees of freedom and is subject to uncertainties including internal modeling errors and external disturbances, operates in complex and dynamic environments. The model is given by

$$\begin{cases} \dot{\eta} = R(\psi)\nu \\ M\ddot{\nu} + C(\nu)\nu + D(\nu)\nu = \tau + d + w \end{cases} \quad (4)$$

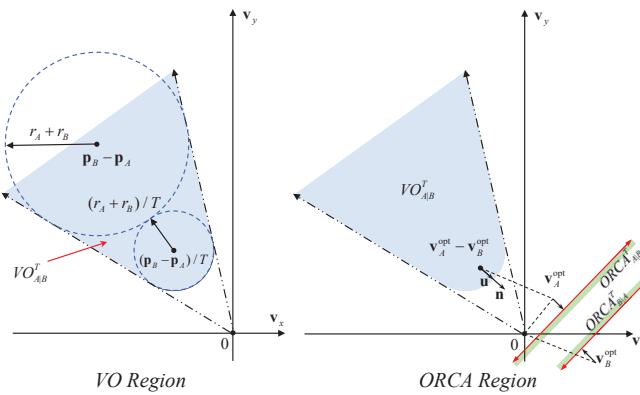


Fig. 1. Visualization of VO and ORCA Regions.

where  $\eta = [x, y, \psi]^T$  is the position-orientation vector, with  $x, y, \psi \in \mathbb{R}$  denoting the surge position, sway position and heading angle, respectively. Similarly,  $\nu = [u, v, r]^T$  is the state vector, with  $u, v, r \in \mathbb{R}$  denoting the surge, sway and yaw velocities, respectively. The term  $\tau = [\tau_u, 0, \tau_r]^T$  is the control input vector, and  $w$  and  $d$  are the unknown modeling error and disturbance, respectively. The inertia, damping, and rotation matrices  $M, C(\nu), D(\nu), R(\psi)$  are modeled as

$$M = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix}, C(\nu) = \begin{bmatrix} 0 & 0 & c_{13} \\ 0 & 0 & c_{23} \\ c_{31} & c_{32} & 0 \end{bmatrix}$$

$$D(\nu) = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & c_{33} \end{bmatrix}, R(\psi) = \begin{bmatrix} \mathbf{c}\psi & -\mathbf{s}\psi & 0 \\ \mathbf{s}\psi & \mathbf{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

where  $m_{11} = m - X_{\dot{u}}, m_{22} = m - Y_{\dot{v}}, m_{23} = m_{32} = mX_{\dot{v}} - Y_{\dot{r}}, m_{33} = I_z - N_{\dot{r}}, c_{13} = -c_{31} = -m_{22}v - \frac{1}{2}(m_{23} + m_{32})r, c_{23} = -c_{32} = m_{11}u, d_{11} = -X_{\dot{u}} - X_{uu}|u|, d_{22} = -Y_{\dot{v}} - Y_{vv}|v| - Y_{rv}|r|, d_{23} = -Y_{\dot{r}} - Y_{vr}|v| - Y_{rr}|r|, d_{32} = -N_{\dot{v}} - N_{vv}|v| - N_{rv}|r|, d_{33} = -N_r - N_{vr}|v| - N_{rr}|r|$ . Besides,  $m$  and  $I_z$  are the USV mass and yaw inertia moment, respectively.  $X_{\dot{u}}, Y_{\dot{v}}, X_{\dot{v}}, N_{\dot{r}}, Y_{\dot{r}}$  are the hydrodynamic derivatives and  $X_u, X_{uu}, Y_v, Y_{vv}, Y_{rv}, Y_r, Y_{vr}, Y_{rr}, N_v, N_{vv}, N_{rv}, N_r, N_{vr}, N_{rr}$  represent the damping coefficients.

### C. Problem statement

The pursuit-encirclement problem of multi-USVs can be decomposed into a velocity planning problem and a velocity tracking problem, specifically involving velocity prediction task, velocity planning task, and control solving task. Considering the complexity of the pursuit-encirclement scenario, which includes uncertainties, incomplete information, and the need for obstacle avoidance, the division into subtasks is necessary. This strategy allows each subtask to pay attention to a specific part of the objective, thereby reducing the interplay among these factors.

For a group of  $K$  USVs performing pursuit-encirclement missions in a complex and dynamic environment, each USV  $m$  with state  $\nu_{t+h}^m = [u_{t+h}^m, v_{t+h}^m, r_{t+h}^m]^T$ , position-orientation  $\eta_{t+h}^m = [x_{t+h}^m, y_{t+h}^m, \psi_{t+h}^m]^T$ , and velocity  $\mathbf{v}_{t+h}^m = [\mathbf{v}_x^m(t+h), \mathbf{v}_y^m(t+h)]^T$  can sense the positions of other mission USVs

and obstacles within observation range  $d_o$  at time  $t+h$ , where  $h$  represents the time interval. Let  $O_m$  be the set of other mission USVs and obstacles observed by USV  $m$ .

- Velocity prediction task: USV  $m$  can observe the position-orientation  $\eta_{t+h}^i$  of the USV  $i$  that falls within its observation range  $d_o$  at time  $t+h$ . The observed  $\eta_{t+h}^i$  is then integrated with the position  $\eta_t^i$  and velocity  $\nu_t^i$  of USV  $i$  at time  $t$ , which are stored in its buffer and equal to zero if  $t=0$  or USV  $i$  is first observed. To utilize this limited information to model the transfer relationship of velocity, a predictive function is employed to reliably predict the velocity of the observed USV at time  $t+h$ :

$$\hat{\nu}_{t+h}^i = \tilde{\Omega}_i(\nu_t^i, \eta_t^i, \eta_{t+h}^i) \quad (6)$$

where  $\tilde{\Omega}_i(\cdot)$  is expressed by NNs.

- Velocity planning task: In the scenario involving  $K$  mission USVs and one evader,  $K$  tracking points are strategically positioned around the evader. The velocities of the mission USVs are planned in such a manner that enables them to reach their respective tracking points as quickly as possible, while ensuring collision avoidance with other USVs and obstacles. The tracking point  $\mathbf{pd}_{t+h}^m = [\mathbf{pd}_x^m(t+h), \mathbf{pd}_y^m(t+h)]^T$  at time  $t+h$ , that has a rotational angular velocity  $\varepsilon$ , is set for USV  $m$  with

$$\mathbf{pd}_{t+h}^m = \mathbf{p}_{t+h}^e + d_{enc} \cdot \begin{bmatrix} \cos \theta_{t+h}^m \\ \sin \theta_{t+h}^m \end{bmatrix} \quad (7)$$

where  $\mathbf{p}_{t+h}^e = [\mathbf{p}_x^e(t+h), \mathbf{p}_y^e(t+h)]^T$ ,  $\theta_{t+h}^m = \theta_t^m + \varepsilon$ , and  $d_{enc}$  are the position of the evader, the encirclement angle, and the encirclement radius, at a time point  $t+h$ .

The preferred velocity  $\mathbf{vp}_{t+h}^m$  for USV  $m$  is determined by selecting the maximum achievable velocity from its current position to the assigned tracking point  $\mathbf{pd}_{t+h}^m$ . This ensures that the USV reaches the tracking point efficiently in a timely manner. Let  $\mathbf{p}_{t+h}^m = [\mathbf{p}_x^m(t+h), \mathbf{p}_y^m(t+h)]^T$  be the position of USV  $m$  at time  $t+h$ . The desired velocity  $\mathbf{vd}_{t+h}^m$  is solved by

$$\begin{aligned} & \arg \min_{\mathbf{v}_{t+h}^m} \|\mathbf{v}_{t+h}^m - \mathbf{vp}_{t+h}^m\|, \\ & \text{s.t. } \|\mathbf{p}_{t+2h}^m - \mathbf{p}_{t+2h}^i\| < r_m + r_i, \forall i \in O_m \end{aligned} \quad (8)$$

where  $r_m, r_i$  denote the radii of USV  $m$  and  $i$ , respectively. The constrained optimization problem (8) is solved in the velocity domain by utilizing the velocity  $\mathbf{v}_{t+h}^m$  and the state  $\hat{\nu}_{t+h}^i$  for each USV  $i$  in the set  $O_m$ .

- Control solving task: The control input  $\tau^m$  of USV  $m$  is solved to influence the position and velocity of the USV, allowing for the adjustment of the velocity of USV  $m$  to track the desired velocity  $\mathbf{vd}_{t+h}^m$ .

The general cost function of USV  $m$  is given by

$$\tilde{J}_m \triangleq \int_0^\infty \Xi(\delta^m) dt \quad (9)$$

where  $\delta^m = \mathbf{v}^m - \mathbf{vd}^m$  represents the velocity tracking error, and  $\Xi(\cdot)$  is a continuous function with respect to its variables. The cost function is optimized to enable the mission USV

to dynamically match the desired velocity profile by making informed control decisions.

As mentioned earlier, the velocity prediction task is crucial to handle the challenges posed by incomplete information, as conventional state prediction/estimation algorithms are not applicable or efficient in this context. The velocity planning task aims to address the obstacle avoidance problem for all mission USVs operating in an environment with multiple observed objects. The control solving task, on the other hand, is designed to achieve optimal control of the USVs. By integrating and collaborating on these three tasks, the pursuit-encirclement mission is accomplished efficiently and safely. Besides, the mission is accomplished through the collaborative efforts of all mission USVs, relying solely on the local observations of each USV, thereby ensuring the deployment of distributed solutions for the mission USVs. Fig. 2 shows the overview of the proposed distributed pursuit-encirclement approach, in which the unmentioned elements are detailed in the following sections.

### III. VELOCITY PLANNING WITH NN-BASED UKF

In this section, NN-based UKF is developed to predict the velocity of USVs with incomplete information. Subsequently, in conjunction with ORCA, velocity planning is completed, ensuring a distributed solution for local observations in complex and dynamic environments.

#### A. NN-based UKF for velocity prediciton

According to the USV dynamics in (4), it can be concluded that

$$\begin{cases} \eta_{t+h} = f(\eta_t, \nu) \\ \nu_{t+h} = g(\nu_t, \nu, \tau) \end{cases} \quad (10)$$

where  $\eta_t$  denotes the USV's position and  $\nu_t$  denotes its velocity at time  $t$ . The measurement equation  $f(\eta_t, \nu)$  and state-transition equation  $g(\nu_t, \nu, \tau)$  are defined as

$$f(\eta_t, \nu) = \eta_t + \int_t^{t+h} R(\psi) \nu dt + W_t \quad (11)$$

$$\begin{aligned} g(\nu_t, \nu, \tau) \\ = \nu_t + \int_t^{t+h} M^{-1}(\tau + d + w - C(\nu)\nu - D(\nu)\nu) dt + H_t \end{aligned} \quad (12)$$

where  $W_t$  and  $H_t$  represent observation noise and process noise, respectively. In the discrete case, the control input  $\tau$  for the USV in (12) can be considered constant over a small time step. Further, it can be assumed that the acceleration  $\dot{\nu}$  remains constant during this time period. Therefore, the measurement equation in (11) can be rewritten as

$$f(\eta_t, \nu_t, \nu_{t+h}) = \eta_t + \frac{1}{2}R(\psi)(\nu_t + \nu_{t+h})h + W_t \quad (13)$$

However, in real-world scenarios involving incomplete information, USVs lack intercommunication, and each USV operates solely based on its local observations. Consequently, a given USV is unable to access the control input  $\tau$  in the state-transition equation  $g(\nu_t, \nu, \tau)$  of other observed USVs, even

within its observation range. To address this limitation, an NN  $\Gamma(\cdot)$  is utilized to approximate the state-transition equation:

$$\Gamma(\cdot) = (W_s)^T \varphi_s(\cdot) \quad (14)$$

where  $(W_s)^T$  with an ideal value of  $(W_s^*)^T$  represents the weight of the network and  $\varphi_s(\cdot)$  represents the activation function. Using NN to approximate the state-transition equation also helps to learn model and environmental features, thereby compensating for the decrease in velocity prediction accuracy caused by modeling errors and environmental disturbances.

Denote the mean and variance of the  $n$ -dimensional state  $\nu$  of the  $i$ th predicted USV as  $\bar{\nu}_t^i$  and  $P_t^i \in \mathbb{R}^{n \times n}$  at time  $t$ , respectively. At  $t = 0$ , assume that  $\bar{\nu}_0^i = \nu_0^i$  and  $P_0^i = 0$ , with  $\nu_0^i$  being the initial state of the USV. Unscented transform (UT) is utilized in the UKF. Calculate  $2n + 1$  sigma points:

$$\nu_{t/t}^{ij} = \begin{cases} \bar{\nu}_t^i, & j = 0 \\ \bar{\nu}_t^i + \left( \sqrt{(n + \lambda)P_t^i} \right)_j, & j = 1 \sim n \\ \bar{\nu}_t^i - \left( \sqrt{(n + \lambda)P_t^i} \right)_{j-n}, & j = n + 1 \sim 2n \end{cases} \quad (15)$$

where  $\left( \sqrt{(n + \lambda)P_t^i} \right)_j$  represents the  $j$ th column of the matrix  $\sqrt{(n + \lambda)P_t^i}$ , and  $\lambda = a^2(n + l) - n$  with hyperparameters  $a, l$  is the proportional adjustment coefficient.

Define the weights of these sigma points  $\omega_t^{ij}$  as

$$\omega_t^{ij} = \begin{cases} \frac{\lambda}{n + \lambda}, & j = 0 \\ \frac{1}{2(n + \lambda)}, & j = 1 \sim 2n \end{cases} \quad (16)$$

To enable the state-transition network  $\Gamma(\cdot)$  to effectively characterize the transition relationships between velocities, the position information of the  $i$ th USV,  $\eta_{t+h}^i$  observed at time  $t+h$ , and  $\eta_t^i$  observed at the previous time  $t$ , are incorporated into the input of the network. This results in the augmented input  $s_{t+h}^{ij} = [(\nu_{t/t}^{ij})^T, (\eta_t^i)^T, (\eta_{t+h}^i)^T]$ , which helps consider additional position transfer features that aid in accurately modeling the state transitions.

Passing the  $2n + 1$  values  $s_{t+h}^{ij}$ ,  $j = 0, 1, \dots, n$ , into the state-transition network  $\Gamma(\cdot)$ , results in obtaining a one-step prediction of  $2n + 1$  sigma points:

$$\nu_{t+h/t}^{ij} = (W_s)^T \varphi_s(s_{t+h}^{ij}) \quad (17)$$

Then, the one-step prediction of the state  $\nu_{t+h}^i$  and its variance matrix  $P_t^i$  is represented as

$$\hat{\nu}_{t+h}^i = \sum_{j=0}^{2n} \omega_t^{ij} \nu_{t+h/t}^{ij} \quad (18)$$

$$P_{t+h/t}^i = \sum_{j=0}^{2n} \omega_t^{ij} (\hat{\nu}_{t+h}^i - \nu_{t+h/t}^{ij})(\hat{\nu}_{t+h}^i - \nu_{t+h/t}^{ij})^T \quad (19)$$

Using predicted values  $\hat{\nu}_{t+h/t}^i$  and  $P_{t+h/t}^i$  to perform UT again, generating new  $2n + 1$  sigma points:

$$\hat{\nu}_{t+h/t}^{ij} = \begin{cases} \hat{\nu}_{t+h}^i, & j = 0 \\ \hat{\nu}_{t+h}^i + \left( \sqrt{(n + \lambda)P_{t+h/t}^i} \right)_j, & j = 1 \sim n \\ \hat{\nu}_{t+h}^i - \left( \sqrt{(n + \lambda)P_{t+h/t}^i} \right)_{j-n}, & j = n + 1 \sim 2n \end{cases} \quad (20)$$

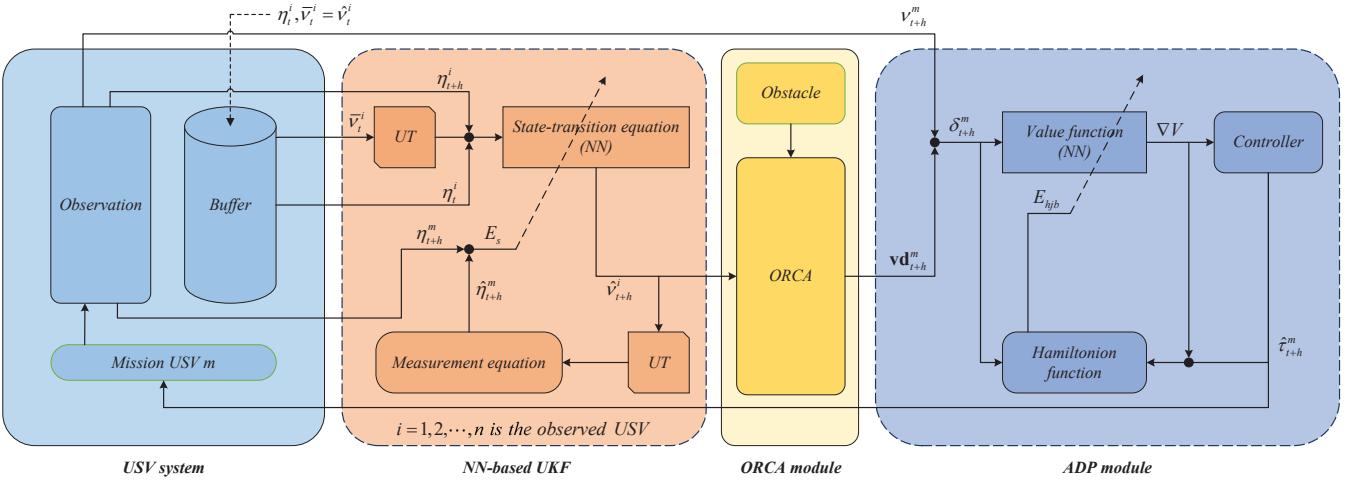


Fig. 2. Overview of the proposed distributed pursuit-encirclement approach.

These newly generated sigma points set, along with the position-orientation  $\eta_t^i$  and state  $\bar{v}_t^i$  from the previous time, are used to generate predictions of  $2n+1$  position-orientations through the measurement equation:

$$\eta_{t+h/t}^{ij} = f(\eta_t^i, \bar{v}_t^i, \hat{v}_{t+h/t}^{ij}) \quad (21)$$

The predicted value and covariance of the position-orientation are obtained through weighting:

$$\hat{\eta}_{t+h}^i = \sum_{j=0}^{2n} \omega_t^{ij} \eta_{t+h/t}^{ij} \quad (22)$$

$$P_{\eta_t \eta_t} = \sum_{j=0}^{2n} \omega_t^{ij} (\hat{\eta}_{t+h}^i - \eta_{t+h/t}^{ij})(\hat{\eta}_{t+h}^i - \eta_{t+h/t}^{ij})^T + \mathfrak{R} \quad (23)$$

$$P_{\nu_t \eta_t} = \sum_{j=0}^{2n} \omega_t^{ij} (\hat{\eta}_{t+h}^i - \eta_{t+h/t}^{ij})(\hat{\eta}_{t+h}^i - \nu_{t+h/t}^{ij})^T \quad (24)$$

where  $\mathfrak{R}$  denotes the covariance matrix of the observation noise  $W_t$ . The Kalman gain is expressed as:

$$K(t+h) P_{\eta_t \eta_t} P_{\nu_t \eta_t}^{-1} \quad (25)$$

Updating the variance  $P_t$  to time  $t+h$ :

$$P_{t+h}^i = P_{t+h/h}^i - K(t+h) P_{\eta_t \eta_t} K^T(t+h) \quad (26)$$

Based on the predicted and true values of the position state, the loss function is defined as

$$E_s = \frac{1}{2} (\eta_{t+h}^i - \hat{\eta}_{t+h}^i)^T (\eta_{t+h}^i - \hat{\eta}_{t+h}^i) \quad (27)$$

By taking the partial derivative of the loss function  $E_s$  with respect to the weight  $W_s$  of network  $\Gamma(s_{t+h}^{ij})$ , denoted as  $\dot{W}_s$ , the weight can be updated using the backpropagation method:

$$W_s = W_s - \alpha_s \dot{W}_s \quad (28)$$

where  $\dot{W}_s = \frac{\partial E_s}{\partial W_s}$ , while  $\alpha_s$  denotes the learning rate. Iteratively train the state-transition network until the weight satisfies  $\|W_s - W_s^*\|_F \leq \sigma_s$  (the standard is considered met when the training error  $E_s$  converges to a sufficiently small

and stable value at any point in time), where  $\sigma_s$  is a reasonable approximate error. Then, consider  $\hat{\eta}_{t+h}^i$  in (18) as the predicted state of the  $i$ th USV observed by USV  $m$ . When it comes to time  $t+2h$ , update  $\bar{v}_{t+h}^i$  with  $\hat{\eta}_{t+h}^i$ .

*Remark 3.1:* By refining the UKF structure and incorporating the NN, the UKF is endowed with the ability to handle incomplete information and perform velocity prediction in complex environments. Based on the NN's adaptive mechanism to the environmental features and USV model features, the prediction accuracy and stability are improved. Furthermore, the straightforward architecture of the NN-based UKF focuses exclusively on handling incomplete information and remains unaffected by control solving, rendering it easily trainable and deployable. In this context, the precise velocity predictions provided by the NN-based UKF serve as a reliable foundation for ORCA to generate collision-free desired velocities for USVs.

### B. ORCA for distributed velocity programming

Within the observation range of USV  $m$ , multiple USVs with incomplete information and obstacles may be present. Ensuring collision avoidance with all observed objects is crucial. In this paper, obstacles are treated as mission USVs with a velocity of 0.

To comply with the operational rules of ORCA, the state  $\nu_{t+h}^i$  of USV  $i$  at time  $t+h$  has to be expressed in terms of vertical velocity and horizontal velocity components. Under the premise of predicted state  $\hat{\eta}_{t+h}^i$ , the predicted velocity of each element in set  $O_m$  at time  $t+h$  is represented as

$$\hat{v}_{t+h}^i = \begin{cases} GR(\psi_{t+h}^i) \hat{\eta}_{t+h}^i, & i \text{ is USV} \\ 0, & i \text{ is obstacle} \end{cases} \quad (29)$$

where  $G = \text{diag}(1, 1, 0)$ .

The collision-free velocity set  $ORCA_{m|i}^T$  of USV  $m$  with respect to the  $i$ th element in  $O_m$  is given in (2), with  $\mathbf{v}_A^{opt} = \mathbf{v}_{t+h}^m, \mathbf{v}_B^{opt} = \mathbf{v}_{t+h}^i$ . The collision-free velocity set of the USV relative to all elements in  $O_m$  is then represented as

$$ORCA_m^T = D(\mathbf{0}, \mathbf{v}_{max}^m) \cap \bigcap_{i \in O_m} ORCA_{m|i}^T \quad (30)$$

where scalar  $\mathbf{v}_{\max}^m$  is the maximum velocity that USV  $m$  can reach. The desired velocity  $\mathbf{vd}_{t+h}^m$ , proximate to the preferred velocity  $\mathbf{vp}_{t+h}^m$ , is selected by

$$\mathbf{vd}_{t+h}^m = \arg \min_{\mathbf{v} \in ORCA_m^T} \|\mathbf{v} - \mathbf{vp}_{t+h}^m\| \quad (31)$$

where  $\mathbf{vp}_{t+h}^m$  is defined as

$$\mathbf{vp}_{t+h}^m = \mathbf{v}_{\max}^m \cdot \begin{bmatrix} \cos \beta \\ \sin \beta \end{bmatrix} \quad (32)$$

with

$$\begin{aligned} \beta &= \\ \text{atan2}(\mathbf{pd}_y^m(t+h) - \mathbf{p}_y^m(t+h), \mathbf{pd}_x^m(t+h) - \mathbf{p}_x^m(t+h)) \end{aligned} \quad (33)$$

The velocity optimization problem in (31) is then solved with 3D Linear programming method, which has been detailed in [15]. Indeed, considering that USVs are subject to nonlinear dynamic constraints as defined in (4), it becomes crucial to efficiently track the desired velocity of each mission USV in complex maritime environments.

Even if the NN within the NN-based UKF is not trained to a satisfactory level and occasionally produces inaccurate velocity predictions, it does not compromise the obstacle avoidance performance of the entire task. This resilience can be attributed to ORCA, which considers collision risk over a future time horizon, thus mitigating the impact of occasional network performance lapses. In essence, as long as the network does not consistently perform poorly, collisions can generally be avoided. Consequently, the velocities planned by ORCA can still be regarded as collision-free.

#### IV. PURSUIT-ENCIRCLEMENT CONTROL WITH ADAPTIVE DYNAMIC PROGRAMMING

In this section, the velocity tracking problem is considered an optimization problem, which, when solved, will drive the USV to complete the pursuit-encirclement task with collision-free guarantee.

##### A. Optimal velocity tracking for the mission USV

The cost function for USV  $m$  in (9) is given at a global level by

$$\begin{aligned} J_m(\tau_t^m, \mathbf{vd}_t^m, \psi_t^m) &= \int_0^\infty \left\{ (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m)^T Q (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m) \right. \\ &\quad \left. + (\tau_t^m)^T F \tau_t^m \right\} dt \end{aligned} \quad (34)$$

where  $Q$  and  $F$  are symmetric positive definite matrices. The term  $\mathbf{GR}(\psi_t^m) \nu_t^m$ , equivalent to  $\mathbf{v}_t^m = [\mathbf{v}_x^m(t), \mathbf{v}_y^m(t)]^T$ , represents the velocity of USV  $m$  at time  $t$ . Using the abbreviation  $\delta_t^m = (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m)$ , the value function of USV  $m$  is defined as

$$V_m(\delta_t^m) = \int_t^\infty \left\{ (\delta_t^m)^T Q \delta_t^m + (\tau_t^m)^T F \tau_t^m \right\} dt \quad (35)$$

The optimization problem analyzed in Section II.C is redefined as an optimization problem for the value function  $V$

at each time point  $t$ , given a desired velocity  $\mathbf{vd}_t^m$ , which is described as

$$\begin{aligned} \min_{\tau_t^m} V_m(\delta_t^m) \\ \text{s.t.} \quad (4) \end{aligned} \quad \text{for given } \mathbf{vd}_t^m \quad (36)$$

The search for optimal control  $\tau_t^m$  at each time steps is accomplished by minimizing the value function  $V_m$  at that time, ensuring the tracking performance of the USV against the desired velocity  $\mathbf{vd}_t^m$ .

##### B. ADP solution for mission USV velocity tracking

To solve the optimization problem of USV  $m$  with respect to its velocity, the Hamiltonian function is given by

$$\begin{aligned} H_m(\delta_t^m, \tau_t^m, \nabla V_m) &\triangleq (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m)^T Q (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m) \\ &\quad + \nabla V_m^T [M^{-1}(\tau_t^m + d^m - C(\nu_t^m) \nu_t^m - D(\nu_t^m) \nu_t^m)] \\ &\quad + (\tau_t^m)^T F \tau_t^m \end{aligned} \quad (37)$$

where  $\nabla V_m = \partial V_m / \partial \nu_t^m = (\partial \delta_t^m / \partial \nu_t^m)^T (\partial V_m / \partial \delta_t^m)$  denotes the partial derivative of the value function  $V_m$  with respect to the state  $\nu_t^m$ . It should be noted that the Hamiltonian function does not explicitly account for incomplete information and obstacle avoidance. However, these aspects are addressed and incorporated within the first two subtasks, which operate at the desired velocity  $\mathbf{vd}_t^m$ . This approach allows for a comprehensive solution without overcomplicating the Hamiltonian function itself.

The optimal value  $\nabla V_m^*$  in Hamiltonian function meets the Hamilton-Jacobi (HJ) equation, which holds

$$0 = \min_{\tau_t^m} H_m(\delta_t^m, \tau_t^m, \nabla V_m^*), \forall i \in M \quad (38)$$

According to the minimum principle [27], the Hamiltonian function in extreme cases follows  $\partial H_m / \partial \tau_t^m = 0$ , i.e.,  $2F\tau_t^m + M^{-1}\nabla V_m = 0$ , from which the optimal control  $\tau_{t^*}^m$  is solved as

$$\tau_{t^*}^m = -\frac{1}{2} F^{-1} M^{-1} \nabla V_m^* \quad (39)$$

Inserting  $\tau_{t^*}^m$  into the HJ equation in (38), it follows that

$$\begin{aligned} &(\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m)^T Q (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m) \\ &+ (\nabla V_m^*)^T [M^{-1}(\tau_{t^*}^m + d^m - C(\nu_t^m) \nu_t^m - D(\nu_t^m) \nu_t^m)] \\ &+ \frac{1}{4} (\nabla V_m^*)^T M^{-1} F^{-1} M^{-1} \nabla V_m^* = 0 \end{aligned} \quad (40)$$

**Theorem 4.1:** Consider the USV system (4). Given any bounded desired velocity  $\mathbf{vd}_t^m$ , the optimal control (39) with the solution  $\nabla V_m^*$  from the HJ equation (40) guarantees that  $\lim_{t \rightarrow \infty} (\mathbf{v}_t^m - \mathbf{vd}_t^m) = 0$ .

*Proof:* Select  $V_m^*$  as the Lyapunov function. Based on the optimal HJ equation presented in (40), the time derivative of  $V_m^*$  is given by

$$\begin{aligned} \dot{V}_m^* &= (\frac{\partial V_m^*}{\partial \nu_t^m})^T \dot{\nu}_t^m \\ &= (\nabla V_m^*)^T [M^{-1}(\tau_{t^*}^m + d^m - C(\nu_t^m) \nu_t^m - D(\nu_t^m) \nu_t^m)] \\ &= -(\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m)^T Q (\mathbf{GR}(\psi_t^m) \nu_t^m - \mathbf{vd}_t^m) \\ &\quad - \frac{1}{4} (\nabla V_m^*)^T M^{-1} F^{-1} M^{-1} \nabla V_m^* \\ &\leq 0 \end{aligned} \quad (41)$$

Note that if  $\delta_t^m = GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m \neq 0$ , then  $\dot{V}_m^* < 0$ , ensuring the convergence of the velocity tracking error, i.e.,  $\lim_{t \rightarrow \infty} \delta_t^m = \lim_{t \rightarrow \infty} (\mathbf{v}_t^m - \mathbf{v}\mathbf{d}_t^m) = 0$ .  $\square$

The value of  $\tau_{t^*}^m$  in (39) relies on the optimal value function  $V_m^*$ , which is difficult to solve in the form of (35) with conventional optimization methods [27], [28]. Therefore, an NN is introduced to represent  $V_m^*$ :

$$V_m^*(\delta_t^m) = (W_c^*)^T \varphi_c(\delta_t^m) + \varepsilon_c(\delta_t^m) \quad (42)$$

where  $W_c^*$ ,  $\varphi_c(\cdot)$  and  $\varepsilon_c(\delta_t^m)$  are the ideal weight, activation function, and the approximation error of the network, respectively. The partial derivative of  $V_m^*$  with respect to  $\nu_t^m$  is represented as

$$\begin{aligned} \nabla V_m^*(\delta_t^m) &= \left( \frac{\partial \delta_t^m}{\partial \nu_t^m} \right)^T \frac{\partial V_m^*}{\partial \delta_t^m} \\ &= GR(\psi_t^m) \left[ \left[ \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} \right]^T W_c^* + \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \right] \end{aligned} \quad (43)$$

By inserting  $\nabla V_m^*$  into (40), the ideal HJ equation is rewritten as

$$\begin{aligned} &H_m(\delta_t^m, \tau_{t^*}^m, \nabla V_m^*) \\ &\triangleq (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m)^T Q (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m) \\ &+ (W_c^*)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T (d^m - C(\nu_t^m)\nu_t^m - D(\nu_t^m)\nu_t^m) \\ &+ \left[ \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \right]^T (Z_t^m)^T (d^m - C(\nu_t^m)\nu_t^m - D(\nu_t^m)\nu_t^m) \\ &- \frac{1}{4} (W_c^*)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T F^{-1} Z_t^m \left[ \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} \right]^T W_c^* \\ &- \frac{1}{4} (W_c^*)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T F^{-1} Z_t^m \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \\ &- \frac{1}{4} \left[ \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \right]^T (Z_t^m)^T F^{-1} Z_t^m \left[ \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} \right]^T W_c^* \\ &- \frac{1}{4} \left[ \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \right]^T (Z_t^m)^T F^{-1} Z_t^m \frac{\partial \varepsilon_c(\delta_t^m)}{\partial \delta_t^m} \\ &= 0 \end{aligned} \quad (44)$$

where  $Z_t^m = M^{-1}GR(\psi_t^m)$ .

Let  $V_m^*(\delta_t^m) = \bar{V}_m(\delta_t^m) + \varepsilon_c(\delta_t^m)$ , where  $\bar{V}_m(\delta_t^m) = (W_c^*)^T \varphi_c(\delta_t^m)$ . It can be deduced from (43) that  $\nabla V_m^*(\delta_t^m) = \nabla \bar{V}_m(\delta_t^m) + GR(\psi_t^m)[\partial \varepsilon_c(\delta_t^m)/\partial \delta_t^m]$ . From (44), the corresponding Hamiltonian function of  $\bar{V}_m(\delta_t^m)$  is denoted as

$$\begin{aligned} &H_m(\delta_t^m, \bar{\tau}_t^m, \nabla \bar{V}_m) \\ &\triangleq (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m)^T Q (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m) \\ &+ (W_c^*)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T (d^m - C(\nu_t^m)\nu_t^m - D(\nu_t^m)\nu_t^m) \\ &- \frac{1}{4} (W_c^*)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T F^{-1} Z_t^m \left[ \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} \right]^T W_c^* \end{aligned} \quad (45)$$

where  $\bar{\tau}_t^m = -\frac{1}{2}F^{-1}M^{-1}\nabla \bar{V}_m$ .

Based on the ADP theory [29], the Hamilton-Jacobi-Bellman (HJB) residual is defined as

$$\varepsilon_{hjb} = H_m(\delta_t^m, \bar{\tau}_t^m, \nabla \bar{V}_m) - H_m(\delta_t^m, \tau_{t^*}^m, \nabla V_m^*) \leq \sigma_{up} \quad (46)$$

where  $\sigma_{up}$  is a positive constant, representing the upper bound of  $\varepsilon_{hjb}$ .

Given that the ideal weight  $W_c^*$  of the optimal value function  $V_m^*$  is unknown,  $\hat{V}_m(\delta_t^m)$  with weight  $\hat{W}_c$  is adopted to approximate it. Denote  $\hat{V}_m(\delta_t^m) = (\hat{W}_c)^T \varphi_c(\delta_t^m)$  and

$\nabla \hat{V}_m(\delta_t^m) = GR(\psi_t^m)[\partial \varphi_c(\delta_t^m)/\partial \delta_t^m]^T \hat{W}_c$ . According to (45), the Hamiltonian function of  $\hat{V}_m(\delta_t^m)$  is given as

$$\begin{aligned} &H_m(\delta_t^m, \hat{\tau}_t^m, \nabla \hat{V}_m) \\ &\triangleq (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m)^T Q (GR(\psi_t^m)\nu_t^m - \mathbf{v}\mathbf{d}_t^m) \\ &+ (\hat{W}_c)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T (d^m - C(\nu_t^m)\nu_t^m - D(\nu_t^m)\nu_t^m) \\ &- \frac{1}{4} (\hat{W}_c)^T \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} (Z_t^m)^T F^{-1} Z_t^m \left[ \frac{\partial \varphi_c(\delta_t^m)}{\partial \delta_t^m} \right]^T \hat{W}_c \end{aligned} \quad (47)$$

where  $\hat{\tau}_t^m = -\frac{1}{2}F^{-1}M^{-1}\nabla \hat{V}_m$ .

Substitute  $H_m(\delta_t^m, \bar{\tau}_t^m, \nabla \bar{V}_m)$  with  $H_m(\delta_t^m, \hat{\tau}_t^m, \nabla \hat{V}_m)$  in (46), then define the approximation error of  $\hat{V}_m$  with

$$e_{hjb} = H_m(\delta_t^m, \hat{\tau}_t^m, \nabla \hat{V}_m) - H_m(\delta_t^m, \tau_{t^*}^m, \nabla V_m^*) \quad (48)$$

By reducing the value of error  $e_{hjb}$ , the disparity between  $H_m(\delta_t^m, \hat{\tau}_t^m, \nabla \hat{V}_m)$  and  $H_m(\delta_t^m, \tau_{t^*}^m, \nabla V_m^*)$  can be minimized. This ultimately leads to the convergence of  $\hat{V}_m$  and  $\hat{\tau}_t^m$  towards their respective optimal values, namely, the optimal value function  $V_m^*$  and the optimal control input  $\tau_{t^*}^m$ .

In order to reduce  $e_{hjb}$ , the objective function is defined as follows:

$$E_{hjb} = \frac{1}{2}e_{hjb}^2 \quad (49)$$

Calculate the partial derivative of  $E_{hjb}$  with respect to the weight  $\hat{W}_c$  associated with  $\hat{V}_m$ . Subsequently, employ backpropagation to update  $\hat{W}_c$  based on the gradient descent method, resulting in the new weight  $\hat{W}_c^{new}$ :

$$\hat{W}_c^{new} = \hat{W}_c - \alpha_c \dot{\hat{W}}_c \quad (50)$$

where  $\dot{\hat{W}}_c = \partial E_{hjb}/\partial \hat{W}_c = e_{hjb}[\partial e_{hjb}/\partial \hat{W}_c]$  and  $\alpha_c$  denotes the learning rate. The control  $\hat{\tau}_t^m$  is considered reliable that can drive the velocity of USV  $m$  to track any given desired velocity  $\mathbf{v}\mathbf{d}_t^m$  when  $\hat{W}_c$  is trained to meet  $\|\hat{W}_c - W_c^*\| \leq \sigma_c$  (the standard is considered met when the training error  $E_{hjb}$  converges to a sufficiently small and stable value at any point in time), where  $\sigma_c$  represents the admissible approximate error.

*Remark 4.1:* The initial two subtasks (Section III) are responsible for managing incomplete information and obstacle avoidance. Consequently, within the ADP framework, the NN is dedicated solely to the optimization of velocity tracking. This exclusive emphasis allows the method to avoid the simultaneous handling of multiple issues in the control solution, simplifying the overall process, speeding up training, enhancing system stability, and ultimately achieving a high level of reliability.

## V. SIMULATION RESULTS

In this section, an example of the pursuit-encirclement control is provided to demonstrate the effectiveness of the proposed method. The multi-USV system consists of 4 mission USVs, with detailed parameters listed in TABLE I. The physical properties of each USV are set as follows: a radius of 1.3m, an observation range of  $d_o = 15$ m, and a maximum navigation speed of 6m/s.

The disturbance  $d$  is generated completely randomly at each time step. The parameters in UKF (15) and (20) are chosen as  $a = 1.2$ ,  $l = 1$ , and the parameters in the cost function (34)

TABLE I  
RELATIVE USV PARAMETERS AND VALUES.

Parameter	Value/Unit	Parameter	Value/Unit
$m$	24.00kg	$Y_{rr}$	-0.35
$I_z$	$1.70\text{kg}\cdot\text{m}^2$	$N_{rr}$	-0.75
$X_u$	-0.63	$N_{rv}$	5.05
$Y_v$	-0.52	$N_{vr}$	0.13
$Y_r$	0.10	$X_v$	0.05
$N_v$	0.10	$Y_r$	0.00
$N_r$	-1.90	$N_r$	-1.00
$X_{uu}$	-1.33	$N_v$	-10.00
$Y_{vv}$	-36.28	$N_{\dot{v}}$	0.00
$Y_{rv}$	-0.81	$X_{\dot{u}}$	-2.00
$Y_{vr}$	-0.85		

are chosen as  $Q = F = \text{diag}(1, 1, 1)$ . Moreover, the learning rate of state-transition NN and value function NN are chosen as  $\alpha_s = \alpha_c = 1e-5$ . The time range for considering collision risk in ORCA is set to 1.2s.

Considering that all mission USVs can only observe local position information of obstacles or other USVs, and there is no communication between mission USVs. Before using the velocity relationship between USVs for pursuit-encirclement control in the velocity domain, the NN-based UKF is first established to predict the velocity of the USV observed by each mission USV. In order to accurately predict the velocity of the observed target in any situation, mission USVs are allowed to roam randomly in the environment to train the state-transition NN. Fig. 3 illustrates the predicted results of the designed NN-based UKF for the surge velocity  $u$ , sway velocity  $v$ , and yaw angular velocity  $r$  of the observed USV. During the entire round of observation of the USV, the maximum prediction errors exhibited are only 0.15m/s, 0.07m/s, and 0.3rad/s. All three velocities closely match their true values, demonstrating the high accuracy, real-time performance, and adaptability to environmental features of the designed NN-based UKF.

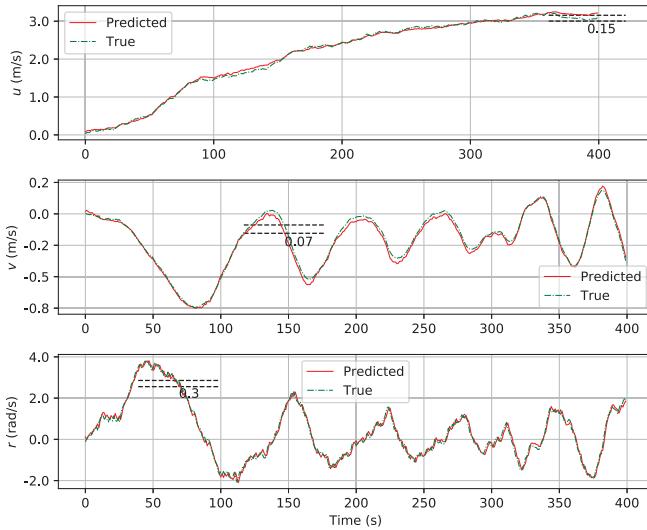


Fig. 3. Velocity predicted results of NN-based UKF.

With the precise velocity prediction of the observed USVs,

the collision-free pursuit-encirclement control for multi-USVs is then completed. Fig. 4 intuitively depicts the situation where 4 mission USVs surround the evaders, with their initial positions being (55m, 55m), (5m, 55m), (5m, 5m), and (55m, 5m). All 4 USVs successfully navigate to the encirclement circle with a radius of 15m without any collisions, and they also steer clear of obstacles encountered during the circumnavigation process. Fig. 5 analyzes the velocity tracking errors of 4 mission USVs for their respective desired velocity in the process shown in Fig. 4. Throughout the entire pursuit-encirclement mission, even if the expected speed changes significantly due to obstacle avoidance behavior, both horizontal and vertical velocity tracking errors are maintained within a reasonable range,  $[-0.23\text{m/s}, 0.14\text{m/s}]$  for horizontal velocity tracking errors and  $[-0.7\text{m/s}, 0.13\text{m/s}]$  for vertical velocity tracking errors. Fig. 6 displays the relative distance changes of four mission USVs towards their respective tracking points in both the vertical and horizontal directions. It can be seen that when obstacle avoidance is necessary, such as encountering environmental obstacles or other mission USVs, the USV momentarily deviates from its tracking point to safely bypass all targets deemed as obstacles. However, in scenarios without any obstacle avoidance requirement, the USV proceeds to reach the tracking point position as quickly as possible.

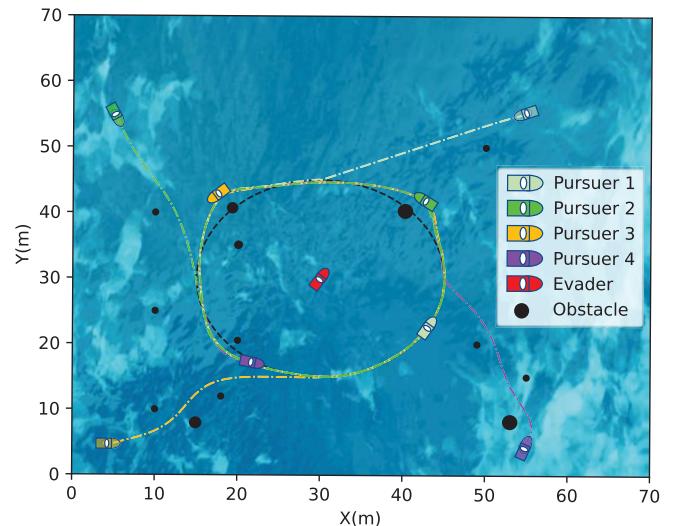


Fig. 4. Collision-free encirclement trajectories of 4 mission USVs.

Fig. 7 depicts the process of the evader attempting to escape, while the mission USVs consistently demonstrate excellent and stable pursuit, obstacle avoidance, and surrounding capabilities. The obstacles depicted in Fig. 7 vary in size, with radii of 0.3m, 0.6m, 0.8m, and 1m. The positions of USVs 1 to 4 in Fig. 7 are respectively initialized as (5m, 5m), (20m, 5m), (5m, 35m), and (5m, 20m), and their surrounding circle radii are set to 10m. To further validate the effectiveness of the proposed approach, experiments using the Robot Operating System (ROS) are also conducted. These experiments are designed to emulate real-world conditions and provide a more comprehensive evaluation of our method. In the established ROS ecosystem, Gazebo, a 3D robotics

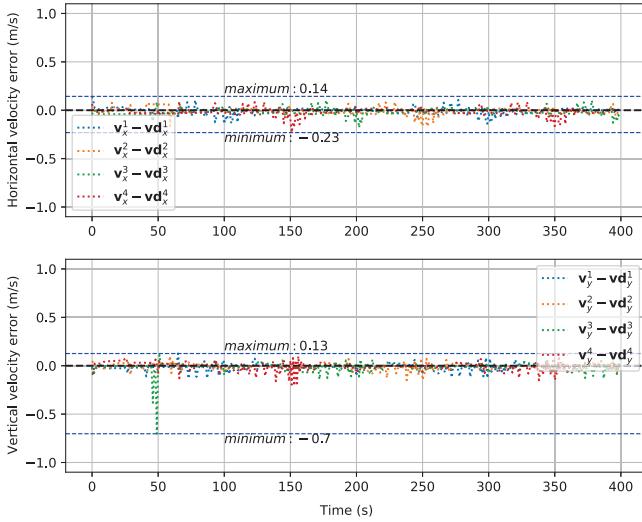


Fig. 5. Velocity tracking errors of 4 mission USVs.

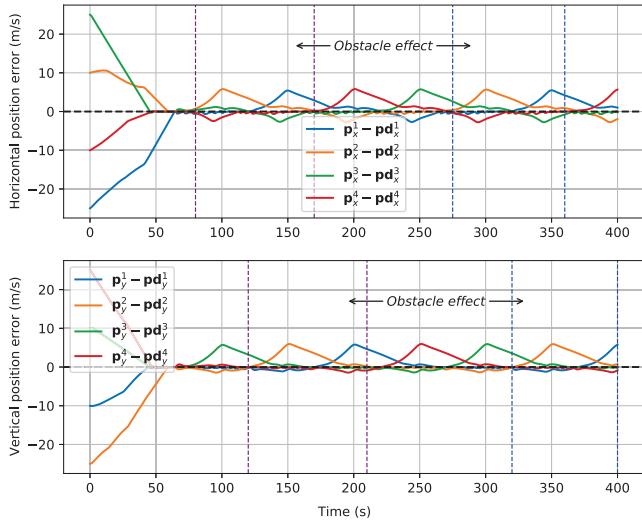


Fig. 6. Position tracking errors of 4 mission USVs.

simulator, is responsible for creating a realistic interaction environment for USVs. Rviz (ROS visualization), on the other hand, is employed to monitor the trajectories and dynamics of the USVs. ROS, acting as the backbone, is not only the underlying control logic, but also the connecting bridge for integrating and coordinating these components. Fig. 8 illustrates the experimental facilities used and the process corresponding to the three stages depicted in Fig. 7 within the Gazebo simulator. The experimental facilities comprise USVs and obstacles categorized as types I to IV, with obstacle radii aligning with the specifications in Fig. 7. In Fig. 9, Rviz depicts the monitored trajectories and dynamics throughout the entire mission process in the Gazebo simulator. Gazebo and Rviz results collectively demonstrate a commendable performance in terms of encirclement and obstacle avoidance, affirming the effectiveness of the proposed method.

Besides digital simulation and software experiments, in order to further validate the performance of the proposed

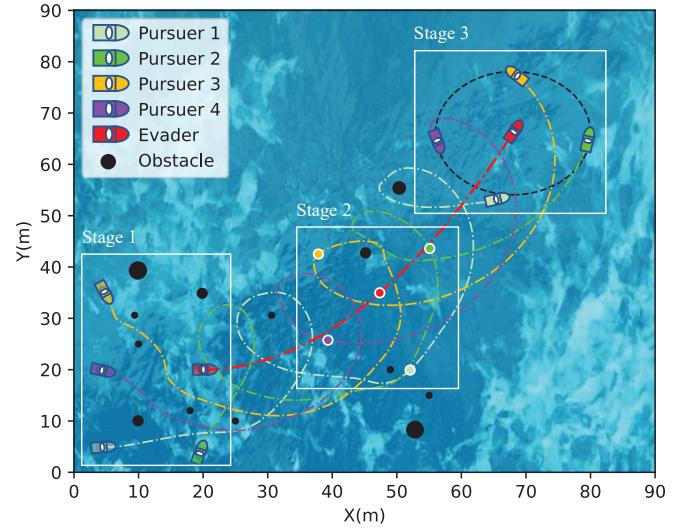


Fig. 7. The pursuit-encirclement trajectories of the 4 mission USVs during the escape of the evader (proposed).

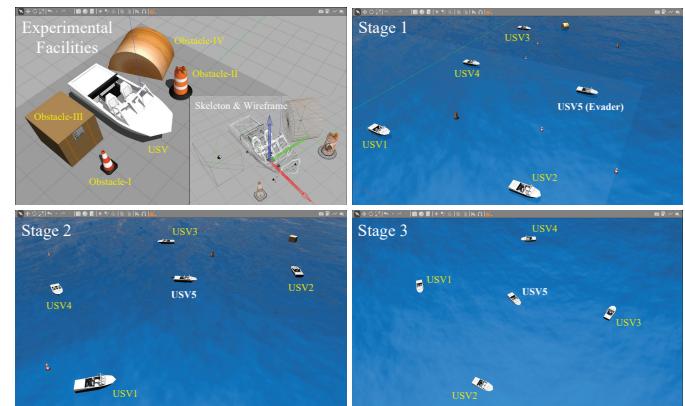


Fig. 8. Experimental facilities and multi-stage mission processes in Gazebo.

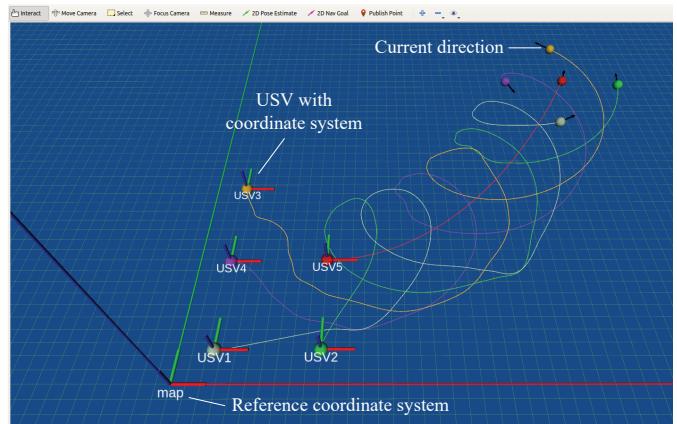


Fig. 9. The trajectories and dynamics monitoring of USVs in Gazebo using Rviz.

method, we compared it with currently outstanding algorithms, namely RL-RVO [30], PPO-max [31], and HATRPO [32], which are suitable for pursuit-encirclement tasks in complex scenarios with incomplete information. Three indicators of

these methods are evaluated separately: (1) Rapidity, which is the average time per round for all USVs to reach their tracking points simultaneously for the first time; (2) Safety, which is  $1 - \text{the proportion of time per round that any USV is within a dangerous distance of } 1.5\text{m from any target}$ ; (3) Encirclement capability, which is the proportion of time per round that all USVs remain within a range of  $1.5\text{m}$  from their tracking points simultaneously. This comparison, detailed in TABLE II, serves to highlight the relative strengths and potential improvements of our approach.

TABLE II  
COMPARISON OF DIFFERENT METHODS

Method	Rapidity	Safety	Encirclement capability
<b>OURS</b>	<b>9.73s</b>	<b>0.99</b>	<b>0.38</b>
RL-RVO	11.69s	0.91	0.31
PPO-max	8.31s	0.85	0.43
HATRPO	8.85s	0.74	0.39

TABLE II illustrates that the proposed method exhibits a significant advantage in terms of safety, a paramount criterion for this task, surpassing other methods. While not consistently the top performer in the other two indicators, the proposed method demonstrates strong competitiveness. To provide a comprehensive comparison, Fig. 10 showcases pursuit-encirclement trajectories generated by the PPO-max algorithm under identical configurations as those in Fig. 7 (representing the proposed method) since PPO-max is the top-performing algorithm among the three compared. In contrast, Fig. 7 exhibits superior real-time tracking, a more stable pursuit process, and a safer obstacle avoidance procedure compared to Fig. 10. As a result, the proposed method consistently excels in terms of overall performance, notably in enhancing the safety of mission-oriented USVs.

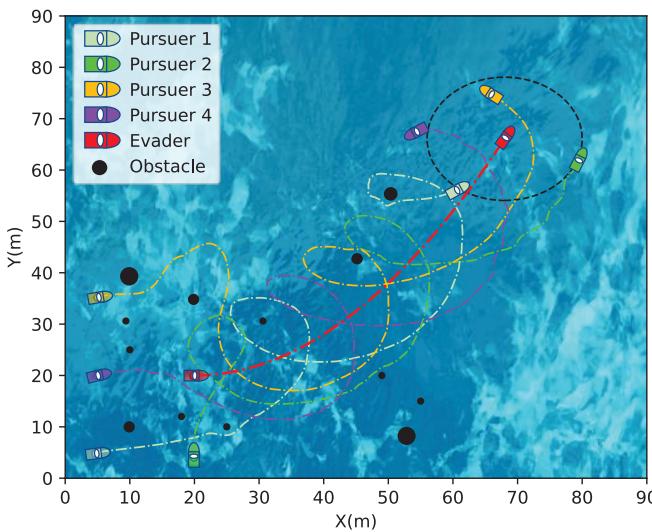


Fig. 10. The pursuit-encirclement trajectories of the 4 mission USVs during the escape of the evader (PPO-max).

The simulation results affirm that the proposed distributed pursuit-encirclement method is capable of coping with complex dynamic environments and accomplishing tasks based

solely on local position observations. The method demonstrates exceptional pursuit efficiency, collision avoidance capability, and encirclement stability in complex and dynamic environments. These results validate the effectiveness and safety of the proposed velocity domain-based distributed approach for multi-USV pursuit-encirclement missions in challenging scenarios.

## VI. CONCLUSION

In this paper, a velocity domain-based distributed approach is proposed to deal with the pursuit-encirclement problem of multi-USVs with incomplete information in a complex environment. In the approach, the developed NN-based UKF is capable of adapting to environmental features by utilizing the adaptive learning and feature extraction capabilities of NNs, which improves the real-time performance and accuracy of the velocity estimation of USVs. ORCA transforms the pursuit-encirclement problem into the velocity domain and balances pursuit and obstacle avoidance behavior to provide the optimal velocity with collision-free guarantee. In order to track the actual velocity of each mission USV to the planned optimal velocity, ADP is used to solve the defined optimization problem. This ensures that the optimal control of multi-USVs can be provided at any time and under any circumstances. Under the cooperation of the decomposed tasks, the approach is easy to train and deploy. The simulation results have shown that the proposed approach has excellent pursuit efficiency, collision avoidance capability, and encirclement stability in pursuit-encirclement mission. Given the inherent limitations imposed by our current hardware infrastructure, our method is currently not feasible for simulation in actual systems. Therefore, our future work includes deploying the proposed method in the real-world environment and extending it to more challenging joint sea-air operations.

## REFERENCES

- [1] B. Liu, H. T. Zhang, H. Meng, D. Fu, and H. Su, "Scanning-chain formation control for multiple unmanned surface vessels to pass through water channels," *IEEE Trans. Cybern.*, vol. 52, no. 3, pp. 1850–1861, 2022.
- [2] P. Wang, K. Yong, M. Chen, and X. Zhang, "Output-feedback flexible performance-based safe formation for underactuated unmanned surface vehicles," *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2437–2447, 2023.
- [3] J. Li, G. Zhang, W. Zhang, Q. Shan, and W. Zhang, "Cooperative Path Following Control of USV-UAVs Considering Low Design Complexity and Command Transmission Requirements," *IEEE Trans. Intell. Veh.*, pp. 1–10, 2023.
- [4] J. Reis, W. Xie, D. Cabecinhas, and C. Silvestre, "Non-linear backstepping controller for an underactuated asv with model parametric uncertainty: Design and experimental validation," *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2514–2526, 2023.
- [5] J. McMahon and E. Plaku, "Autonomous data collection with dynamic goals and communication constraints for

- marine vehicles,” *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 3, pp. 1607–1620, 2023.
- [6] X. Wang, C. Jiang, S. Sheng, and Y. Xu, “Stop-line-aided cooperative positioning for connected vehicles,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1765–1776, 2023.
  - [7] B. Bovcon, J. Muhovič, D. Vranac, D. Mozetič, J. Perš, and M. Kristan, “Mods—a usv-oriented object detection and obstacle segmentation benchmark,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13403–13418, 2022.
  - [8] F. M. Ortiz, M. Sammarco, L. H. M. K. Costa, and M. Detyniecki, “Applications and services using vehicular exteroceptive sensors: A survey,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 949–969, 2023.
  - [9] J. Fu, W. Yao, G. Sun, H. Tian, and L. Wu, “An ftsa trajectory elliptical homotopy for unmanned vehicles path planning with multi-objective constraints,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2415–2425, 2023.
  - [10] R. Han, S. Chen, S. Wang, Z. Zhang, R. Gao, Q. Hao, and J. Pan, “Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards,” *IEEE Rob. Autom. Lett.*, vol. 7, no. 3, pp. 5896–5903, 2022.
  - [11] H. Li, Y. Cai, J. Hong, P. Xu, H. Cheng, X. Zhu, B. Hu, Z. Hao, and Z. Fan, “Vg-swarm: A vision-based gene regulation network for uavs swarm behavior emergence,” *IEEE Rob. Autom. Lett.*, vol. 8, no. 3, pp. 1175–1182, 2023.
  - [12] Z. Chu, F. Wang, T. Lei, and C. Luo, “Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 108–120, 2023.
  - [13] Z. Sun, H. Sun, P. Li, and J. Zou, “Self-organizing cooperative pursuit strategy for multi-usv with dynamic obstacle ships,” *J. Mar. Sci. Eng.*, vol. 10, no. 5, 2022.
  - [14] Z. Qi, T. Wang, J. Chen, D. Narang, Y. Wang, and H. Yang, “Learning-based path planning and predictive control for autonomous vehicles with low-cost positioning,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1093–1104, 2023.
  - [15] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Proc. Int. Symp. Robot. Res.*, 2009.
  - [16] K. Guo, D. Wang, T. Fan, and J. Pan, “Vr-orca: Variable responsibility optimal reciprocal collision avoidance,” *IEEE Rob. Autom. Lett.*, vol. 6, no. 3, pp. 4520–4527, 2021.
  - [17] F. Ho, R. Geraldes, A. Gonçalves, M. Cavazza, and H. Prendinger, “Improved conflict detection and resolution for service uavs in shared airspace,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1231–1242, 2019.
  - [18] J. Bertram, P. Wei, and J. Zambreno, “A fast markov decision process-based algorithm for collision avoidance in urban air mobility,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15420–15433, 2022.
  - [19] N. Gu, D. Wang, Z. Peng, and L. Liu, “Observer-based finite-time control for distributed path maneuvering of underactuated unmanned surface vehicles with collision avoidance and connectivity preservation,” *IEEE Trans. Syst. Man Cybern.: Syst.*, vol. 51, no. 8, pp. 5105–5115, 2021.
  - [20] G. Qi, X. Li, and Z. Chen, “Problems of extended state observer and proposal of compensation function observer for unknown model and application in uav,” *IEEE Trans. Syst. Man Cybern.: Syst.*, vol. 52, no. 5, pp. 2899–2910, 2022.
  - [21] J. Ghommam, M. Saad, F. Mnif, and Q. M. Zhu, “Guaranteed performance design for formation tracking and collision avoidance of multiple usvs with disturbances and unmodeled dynamics,” *IEEE Syst. J.*, vol. 15, no. 3, pp. 4346–4357, 2021.
  - [22] E. Wan and R. Van Der Merwe, “The unscented kalman filter for nonlinear estimation,” in *Proc. IEEE Adaptive Syst. Signal Process., Commun., Control Symp.*, 2000, pp. 153–158.
  - [23] P. Dahal, S. Mentasti, S. Arrigoni, F. Braghin, M. Matteucci, and F. Cheli, “Extended object tracking in curvilinear road coordinates for autonomous driving,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1266–1278, 2023.
  - [24] X. Cui and B. Xu, “State of charge estimation of lithium-ion battery using robust kernel fuzzy model and multi-innovation ukf algorithm under noise,” *IEEE Trans. Ind. Electron.*, vol. 69, no. 11, pp. 11121–11131, 2022.
  - [25] H. Peng, B. Zeng, L. Yang, Y. Xu, and R. Lu, “Distributed extended state estimation for complex networks with nonlinear uncertainty,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 34, no. 9, pp. 5952–5960, 2023.
  - [26] L. Xue, B. Ma, J. Liu, C. Mu, and D. C. Wunsch, “Extended kalman filter based resilient formation tracking control of multiple unmanned vehicles via game-theoretical reinforcement learning,” *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2307–2318, 2023.
  - [27] K. G. Vamvoudakis, F. L. Lewis, and G. R. Hudas, “Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality,” *Automatica*, vol. 48, no. 8, pp. 1598–1611, 2012.
  - [28] K. Ito, *Optimal Control*, pp. 1–20. John Wiley & Sons, Ltd, 2017.
  - [29] Y. Yang, W. Gao, H. Modares, and C. Z. Xu, “Robust actor-critic learning for continuous-time nonlinear systems with unmodeled dynamics,” *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 6, pp. 2101–2112, 2022.
  - [30] R. Han, S. Chen, S. Wang, Z. Zhang, R. Gao, Q. Hao, and J. Pan, “Reinforcement Learned Distributed Multi-Robot Navigation With Reciprocal Velocity Obstacle Shaped Rewards,” *IEEE Rob. Autom. Lett.*, vol. 7, no. 3, pp. 5896–5903, 2022.
  - [31] R. Zheng, S. Dou, S. Gao, and et al., “Secrets of RLHF in Large Language Models Part I: PPO,” 2023, doi: <https://doi.org/10.48550/arXiv.2307.04964>
  - [32] J. G. Kuba, R. Chen, M. Wen, and et al., “Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning,” 2022, doi: <https://doi.org/10.48550/arXiv.2109.11251>.



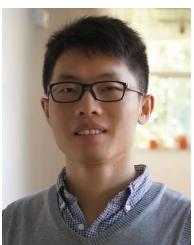
**Yangyang Liu** received the B.S. degree in automation from the School of Nanjing Institute of Technology, Nanjing, China, in 2022. He is currently pursuing the M.S. degree with the School of Mechatronic Engineering and Automation, Shanghai University. His research interests include deep reinforcement learning, game theory and pursuit-evasion game for multi-agent systems and their applications.



**Chun Liu** received the Ph.D. degree in control theory and engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2020. He is currently an associate professor with the School of Mechatronic Engineering and Automation and with the Institute of Artificial Intelligence, Shanghai University, China. His research interests include fault diagnosis and fault tolerant control for multi-agent systems and their applications.



**Yizhen Meng** received the Ph.D. degree in control theory and engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2020. He is currently working at the Shanghai Aerospace Control Technology Institute, China. His research interests include multi-agent reinforcement learning games, cooperation, and their applications in multi-agent cluster obstacle avoidance.



**Xiaoliang Ren** received the B.E. degree in Automation from Zhejiang University, Hangzhou, China, in 2012 and the Ph.D. degree in control and dynamic systems from Hong Kong University of Science and Technology in 2016. Prior to his current position, he was a postdoctoral researcher in the Hong Kong University of Science and Technology in 2016, Nanyang Technological University from 2016 to 2018, and KTH Royal Institute of Technology from 2018 to 2019. He is currently a professor at the School of Mechatronic Engineering and Automation, Shanghai University, China. His research interests include control, optimization, and security of cyber-physical systems.



**Xiaofan Wang** received the Ph.D. degree from Southeast University, China, in 1996. He has been a Professor with Shanghai Jiao Tong University (SJTU) since 2002 and a Distinguished Professor of SJTU since 2008. He is now the President of Shanghai Institute of Technology. He received the 2002 National Science Foundation for Distinguished Young Scholars of P. R. China, the 2005 Guillemin-Cauer Best Transactions Paper Award from the IEEE Circuits and Systems Society, the 2008 Distinguished Professor of the Chang Jiang Scholars Program, Ministry of Education, and the 2015 Second Class Prize of the State Natural Science Award. His current research interests include analysis and control of complex dynamical networks. He is currently vice chair of IFAC Coordinating Committee CC5 and vice-chair of Systems Engineering Society of China.