MOTION CONTROL AND COLLISION AVOIDANCE ALGORITHMS FOR UNMANNED SURFACE VEHICLE SWARM IN PRACTICAL MARITIME ENVIRONMENT

Jiayuan Zhuang[1]
Lei Zhang[1]
Zihe Qin[2]
Hanbing Sun[1]
Bo Wang[1]
Jian Cao[1]
[1] College of Shipbuilding Engineering, Harbin Engineering University, China
[2] Zhuhai Yunzhou Intelligence Technology Ltd., China

ABSTRACT

The issue of controlling a swarm of autonomous unmanned surface vehicles (USVs) in a practical maritime environment is studied in this paper. A hierarchical control framework associated with control algorithms for the USV swarm is proposed. In order to implement the distributed control of the autonomous swarm, the control framework is divided into three task layers. The first layer is the tele-operated task layer, which delivers the human operator's command to the remote USV swarm. The second layer deals with autonomous tasks (i.e. swarm dispersion, or avoidance of obstacles and/or inner-USV collisions), which are defined by specific mathematical functions. The third layer is the control allocation layer, in which the control inputs are designed by applying the sliding mode control method. The motion controller is proved asymptotically stable by using the Lyapunov method. Numerical simulation of USV swarm motion is used to verify the effectiveness of the control framework.

Keywords: USV; swarm control; collision avoidance; sliding mode; artificial potential field.

INTRODUCTION

In recent years, the concept of marine ranching has become a research hotspot. However, in a real case, it is a high risk that the marine ranching may get stolen, and more than ten thousand dollars can be lost in a single night. Thus, the surveillance and observation performed by a number of cooperating USVs is of great practical significance. Beside the surveillance task and patrolling the growth environment of marine creatures, it is also sometimes necessary to achieve the roundup of thieving ships.

The USV motion control has received high attention from control communities in the past decades, and the research topics about USVs were focused on path planning, path following, and formation keeping [1-3]. Recently, there are two significant trends in research focused on USV motion control. One is to equip the control system with intelligence, or in other words to achieve autonomous control of USV, while the other one is to extend the autonomous control from a single vessel to a number of cooperating vessels.

So far, studies on multi-agents are mainly about formation keeping, obstacle avoidance, and connectivity maintaining [4-6]. For the formation keeping, extensive studies have been made, and the study objects include satellites, aircrafts, vessels, and underwater vehicles. During the research progress, three dominating methods for formation control have been universally acknowledged, i.e. the leader-follower method [7], the virtual structure method [8-9], and the behavior based

method [10]. The formation control can preserve the desired formation well, but the controller is centralized, and the members are not endowed with the capability of decision-making. Thus, when dealing with some problems, for instance avoiding obstacles and/or collisions with other fleet's members, the controller might get unstable [11].

It is worth mentioning that the concept of swarm control has become a focus recently. It aims to achieve automatic control of a multi-agent system [12-13]. Being superior to the formation control, the swarm control focuses on attaching robotics with intelligence. Much of the inspiration for artificial swarms stems from the study of biological swarms, and most of the earlier investigations that focus on this general theme of artificial swarm consider a homogenous swarm [14, 15]. In Cepeda-Gomez et al. [14], a robust sliding mode control strategy is implemented on two competing multi-agent swarms, called pursuers and evaders. The proposed control achieves the stability and the ability to capture the evaders by the pursuers despite the uncertainties in the evaders' behavior. Liu et al. [16] have studied the problem of bilateral human-swarm interaction, which enables a human operator to simultaneously interact with a group of swarm robots in a remote environment. The authors have developed a task-oriented control framework for 2-DOF robots.

So far, swarm research is mainly about ground robots and unmanned aerial vehicles. In general, USV swarm is the object of research in the aspects of obstacle avoidance, formation control, and collision avoidance.

In this paper, the swarm control problem of a group of autonomous USVs is studied, and a hierarchical control framework is proposed. The underactuated property is considered in the controller design, and control allocation is then supplemented. According to different functions of tele-operated task, autonomous tasks, and control allocation, the designed controller has three layers. The controller is proved asymptotically stable by using the Lyapunov method. Numerical simulation is also provided to illustrate the validity and effectiveness of the proposed method.

USV DYNAMICS

In motion control of ships in the horizontal plane, it is conventional to ignore heave, pitch, and roll motions, and only consider such motions as surge, sway and yaw. Assume that the swarm is composed of n USVs, and for the ith USV, the mathematical equations of kinematics and kinetics are defined as follows [17]:

$$\dot{\boldsymbol{\eta}}_i = \boldsymbol{J}_i(\psi_i)\boldsymbol{v}_i$$
$$\boldsymbol{M}_i\dot{\boldsymbol{v}}_i = -\boldsymbol{C}(\boldsymbol{v}_i)\boldsymbol{v}_i - \boldsymbol{D}_i\boldsymbol{v}_i + \boldsymbol{\tau}_i \tag{1}$$

where $\boldsymbol{\eta}_i = (x_i, y_i, \psi_i)^T$ denotes the earth-fixed coordinate frame coordinates and the yaw angle of ith USV, and $\boldsymbol{v}_i = (u_i, v_i, r_i)^T$ denotes the body fixed velocities for $i = 1, 2, \cdots, n$. $\boldsymbol{J}_i(\psi_i)$ denotes the rotation matrix of the ship

from body-fixed frame to earth-fixed frame. The frame transformation is presented in Fig. 1, where the X-axis points to the north and the Y-axis points to the east.
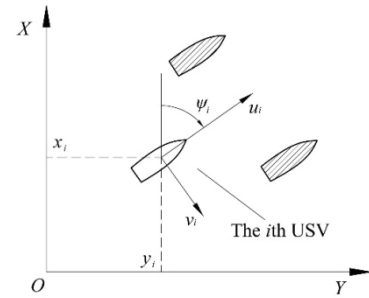

Fig. 1. Frame transformation in horizontal plane

$\boldsymbol{M}_i$ is the inertia matrix considered with hydrodynamic added inertia. $\boldsymbol{C}(\boldsymbol{v}_i)$ denotes the Coriolis and centripetal matrix, which is caused by the rotation. $\boldsymbol{D}_i$ denotes the damping matrix, which is composed of damping coefficients in the surge, sway, and yaw axis directions. $\boldsymbol{\tau} = (\tau_u, 0, \tau_r)^T$ is the control input, which consists of surge thrust and yaw moment. It should be pointed out that $\boldsymbol{M}_i$ and $\boldsymbol{D}_i$ are both positive definite, and they are assumed to be constant. The abovementioned matrixes have the following forms:

$$\boldsymbol{M}_i = \begin{bmatrix} m_{11i} & 0 & 0 \\ 0 & m_{22i} & 0 \\ 0 & 0 & m_{33i} \end{bmatrix}, \ \boldsymbol{C}(\boldsymbol{v}_i) = \begin{bmatrix} 0 & 0 & -m_{22i}v_i \\ 0 & 0 & m_{11i}u_i \\ m_{22i}v_i & -m_{11i}u_i & 0 \end{bmatrix},$$

$$\boldsymbol{D}_i = \begin{bmatrix} d_{ui} & 0 & 0 \\ 0 & d_{vi} & 0 \\ 0 & 0 & d_{ri} \end{bmatrix}, \ \boldsymbol{J}_i(\psi_i) = \begin{bmatrix} \cos\psi_i & -\sin\psi_i & 0 \\ \sin\psi_i & \cos\psi_i & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Based on the motion model of each individual, the integral dynamics of the swarm can be defined as:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\psi)\boldsymbol{v}$$
$$\boldsymbol{M}\dot{\boldsymbol{v}} = -\boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} - \boldsymbol{D}\boldsymbol{v} + \boldsymbol{\tau} \tag{2}$$

where the symbols in Eq. (2) have the following form:

$$\boldsymbol{\eta} = [\boldsymbol{\eta}_1^T, \boldsymbol{\eta}_2^T, \cdots, \boldsymbol{\eta}_n^T]^T, \ \boldsymbol{v} = [\boldsymbol{v}_1^T, \boldsymbol{v}_2^T, \cdots, \boldsymbol{v}_n^T]^T, \ \boldsymbol{\tau} = [\boldsymbol{\tau}_1^T, \boldsymbol{\tau}_2^T, \cdots, \boldsymbol{\tau}_n^T]^T,$$

$$\boldsymbol{J} = diag\{\boldsymbol{J}_1, \boldsymbol{J}_2, \cdots, \boldsymbol{J}_n\}, \ \boldsymbol{M} = diag\{\boldsymbol{M}_1, \boldsymbol{M}_2, \cdots, \boldsymbol{M}_n\},$$

$$\boldsymbol{C}(\boldsymbol{v}) = diag\{\boldsymbol{C}(\boldsymbol{v}_1), \boldsymbol{C}(\boldsymbol{v}_2), \cdots, \boldsymbol{C}(\boldsymbol{v}_n)\} \boldsymbol{D} = diag\{\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_n\}.$$

FRAMEWORK DESIGN FOR SWARM CONTROL

In order to achieve the autonomous motion control of USV swarm, the controller is bilaterally designed according to different goals. It is composed of three layers, the supreme of which is the tele-operated task layer. In this layer, the center

position of the swarm and the average distance between USVs and the center position are remotely controlled by a human operator. The second layer is the autonomous task design, in which some autonomous tasks are defined to solve specific problems in practical navigation, and it is essential for these tasks to meet the demand of the tele-operation layer. The bottom layer is the control allocation design. In this layer, the underactuated feature of USVs is considered, and the control input is re-designed to fit the drive appliances.

TELE-OPERATED TASK LAYER

In this layer, an individual position of each USV is not directly controlled. The human operator only controls the center position and the average distance, the desired values of which are delivered to each USV. In order to express these orders in a mathematical way, the desired position function is defined as follows:

$$X_h = [\overline{x}_d, \overline{y}_d, \sigma_d]^T \tag{3}$$

where $(\overline{x}_d, \overline{y}_d)$ denotes the desired value of the center position and σ_d denotes the desired distance variance.

According to the form of Eq. (3), the position function of the swarm is defined as:

$$X_s = [\overline{x}, \overline{y}, \sigma]^T \tag{4}$$

where $(\overline{x}, \overline{y})$ denotes the desired value of the center position, and σ denotes the desired distance variance, which can be written as follows:

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad \overline{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

$$\sigma = \frac{1}{n^2} (\sum_{i=1}^{n} [(x_i - \overline{x})^2 + (y_i - \overline{y})^2]) \tag{5}$$

The tele-operated task error e_h is further defined as:

$$e_h = X_s - X_h = [e_1, e_2, e_3]^T \tag{6}$$

where e_1, e_2, and e_3 denote the components of e_h.

The control framework with the task layer design and the sequence of information transfer is shown in Fig. 2. Some symbols shown in the figure, such as η and τ', will be explained later in the article.

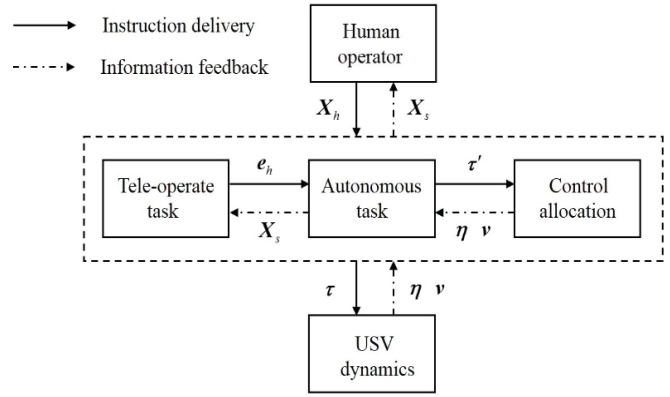*Fig. 2. Framework of the closed-loop control system*

Differentiating Eq. (4) we get

$$\dot{X}_s = J_s(\eta)\dot{\eta} \tag{7}$$

where J_s is the Jacobian matrix having the following form:

$$J_s(\eta) = \frac{\partial X_s}{\partial \eta} = \left[\frac{\partial X_{s1}}{\partial \eta}^T, \frac{\partial X_{s2}}{\partial \eta}^T, \frac{\partial X_{s3}}{\partial \eta}^T \right]^T \in R^{3 \times 3n} \tag{8}$$

Notice that J_s is not a square matrix, and the pseudo-inverse of J_s is defined as $J_s^+ = J_s^T(J_s J_s^T)$, which satisfies the condition of $J_s J_s^+ = I_3$ [18].

AUTONOMOUS TASK LAYER

During the autonomous navigation of USV swarm, many autonomous tasks may need to be conducted simultaneously, and controlled solely by USVs instead of the human operator. To simplify the research, only 3 typical cases have been selected as autonomous tasks, which are: obstacle avoidance, inner USV collision avoidance, and swarm dispersion. The autonomous tasks are designed by making use of potential functions. By defining a specific mathematical function for each task, the corresponding purposes can be achieved.

Obstacle avoidance

In this task the obstacles are assumed to be known, and the collision avoidance algorithm presented by Stipanović et al. [19] for a multi-agent system is used for the USV swarm. Then the obstacle avoidance task function Φ_{oj} is given by

$$\Phi_{oj} = \sum_{i=1}^{n} \{ \min(0, \frac{d(i,o_j)^2 - R_o^2}{d(i,o_j)^2 - r_o^2}) \}^2 \tag{9}$$

where $d(i,o_j)$ is the distance between the ith USV and the jth obstacle, whose position in the remote environment is

denoted by η_{oj} and given by the operator. R_o is the avoidance distance, and r_o is the shortest safe distance. As for Eq. (9), it can be used to avoid both static and moving obstacles. Thus, the avoidance of sailing ships can be achieved as well.

The partial derivative of Φ_{oj} for η has the following form

$$\frac{\partial \Phi_{oj}}{\partial \boldsymbol{\eta}} = 4 \frac{(R_o^2 - r_o^2)(d(i,o_j)^2 - R_o^2)}{(d(i,o_j)^2 - r_o^2)^3} (\boldsymbol{\eta}_s - \boldsymbol{\eta}_{oj})^T \text{ , if } r_o < d(i,o_j) < R_o \quad \text{(10)}$$

where $\boldsymbol{\eta}_s = [x_1, y_1, 0, \cdots, x_n, y_n, 0,]^T$ denotes the position of USVs, $\boldsymbol{\eta}_{oj} = [ox_j, oy_j, 0, \cdots, ox_j, oy_j, 0]^T$ is the obstacle vector, and $\boldsymbol{\eta}_s, \boldsymbol{\eta}_{oj} \in R^{3n}$.

The auxiliary control function φ_o is further defined as:

$$\boldsymbol{\varphi}_o = -\sum_{j=1}^{m} (\partial \Phi_{oj} / \partial \boldsymbol{\eta})^T \quad \text{(11)}$$

Inner USV collision avoidance

This task aims to keep a safe distance between each pair of USVs to avoid collision between them. Similarly to Eq. (9), the inner USV collision avoidance can be achieved by defining the following function:

$$\Phi_{rj} = \sum_{i=1}^{n} \{\min(0, \frac{d(i,j)^2 - R_r^2}{d(i,j)^2 - r_r^2})\}^2 \quad \text{(12)}$$

where $d(i,j)$ is the distance between the ith USV and the jth USV, R_r is the avoidance distance, and r_r is the smallest safe distance between two USVs.

Taking the partial derivative of Φ_r for η we get the following equation:

$$\frac{\partial \Phi_{rj}}{\partial \boldsymbol{\eta}} = 4 \frac{(R_r^2 - r_r^2)(d(i,j)^2 - R_r^2)}{(d(i,j)^2 - r_r^2)^3} (\boldsymbol{\eta}_s - \boldsymbol{\eta}_{rj})^T, \text{ if } r_r < d(i,j) < R_r \quad \text{(13)}$$

where $\boldsymbol{\eta}_{rj} = [x_j, y_j, 0, \cdots, x_j, y_j, 0]^T$ is the position vector for the ith USV, and $\boldsymbol{\eta}_{rj} \in R^{3n}$

The auxiliary control function φ_{rj} is further defined as:

$$\boldsymbol{\varphi}_r = -\sum_{j=1}^{m} (\partial \Phi_{rj} / \partial \boldsymbol{\eta})^T \quad \text{(14)}$$

Swarm dispersion

USVs might gather together into several small groups while achieving the tele-operated task. However, in applications such as coverage control of a mobile robot network, increasing the coverage area is necessary to improve the swarm performance in the cases of surface search or seabed detection [20]. The swarm dispersion task is considered to maximize the inter-member distance and thus to enlarge the area of coverage. The task function is given as

$$\Phi_d = -\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d(i,j)^2 \quad \text{(15)}$$

The auxiliary control function φ_{rj} is further defined as

$$\boldsymbol{\varphi}_d = -(\partial \Phi_d / \partial \boldsymbol{\eta})^{\mathrm{T}} = \sum_{j=1}^{n} (\boldsymbol{\eta}_s - \boldsymbol{\eta}_{rj})^T \quad \text{(16)}$$

Remark 1. In the autonomous task layer, the abovementioned tasks should be considered simultaneously. The autonomous task functions can be combined to generate the auxiliary function, for example $\Phi = k_1 \Phi_{oj} + k_2 \Phi_{rj} + k_3 \Phi_d$. Then the total auxiliary control function is given as $\boldsymbol{\varphi} = k_1 \boldsymbol{\varphi}_o + k_2 \boldsymbol{\varphi}_r + k_3 \boldsymbol{\varphi}_d$, where k_1, k_2 and k_3 are proper gains for each task.

As for parameters k_1, k_2 and k_3, they are proportional to relevant virtual forces, and they can be chosen according to task priority. The obstacle avoidance task has higher priority than inner USV collision avoidance, and swarm dispersion is the last task needed to fulfill. In addition, virtual repulsive forces for each USV should be large enough to achieve the goal of collision avoidance, but also should not be excessively large to avoid unstable operation of the controller system. Combining the above factors, k_1 and k_2 were assumed to be large constants, and k_3 to be a large constant at the beginning. Then multiple simulation tests were conducted, after which the final values of parameters were selected as k_1=20, k_2=5, k_3=0.1.

Preliminary design of control input

To accomplish the tele-operated task and autonomous task, the following Lyapunov function is defined:

$$V_1 = \frac{1}{2} e_h^T e_h + \Phi \quad \text{(17)}$$

Differentiating both sides of Eq. (17), we get

$$\dot{V}_1 = e_h^T \dot{e}_h - \lambda \boldsymbol{\varphi}^T \dot{\boldsymbol{\eta}} = e_h^T (J_s \dot{\boldsymbol{\eta}} - \dot{X}_h) - \boldsymbol{\varphi}^T \dot{\boldsymbol{\eta}}$$
$$= (e_h^T J_s - \boldsymbol{\varphi}^T) \dot{\boldsymbol{\eta}} - e_h^T \dot{X}_h \quad \text{(18)}$$

According to Eq. (18), we define a virtual control item to make V_1 converged, which is designed as follows:

$$\dot{\boldsymbol{\eta}}^\alpha = -k_4 (J_s^T e_h - \boldsymbol{\varphi}) \quad \text{(19)}$$

where $\dot{\boldsymbol{\eta}}^\alpha$ is the desired value of $\dot{\boldsymbol{\eta}}$, and k_4 is a positive constant coefficient. Then we define another error variable as: $\boldsymbol{\xi} = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}^\alpha$.

In the task design, it is convenient to set X_h constant over a certain period of time, as a result of which $\dot{X}_h = 0$. By substituting Eq. (19) into Eq. (18), $\dot{V}_1$ can be rewritten as:

$$\dot{V}_1 = (e_h^T J_s - \boldsymbol{\varphi}^T)(\dot{\boldsymbol{\eta}}^\alpha + \boldsymbol{\xi})$$
$$= -k_4 (J_s^T e_h - \boldsymbol{\varphi})^T (J_s^T e_h - \boldsymbol{\varphi}) + (J_s^T e_h - \boldsymbol{\varphi})^T \boldsymbol{\xi} \quad \text{(19)}$$

In the next step, another Lyapunov function V_2 is defined as

$$V_2 = V_1 + \frac{1}{2}\boldsymbol{\xi}^T \boldsymbol{\xi} \tag{20}$$

Taking the time derivative of V_2 along the solution of Eq. (19) and setting $\boldsymbol{J\dot{v}} = -k_5\boldsymbol{\xi} - \boldsymbol{\dot{J}v}$, where k_4 is also a positive coefficient, we get Eq. (21) as follows:

$$\dot{V}_2 = -k_4(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi})^T(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi}) - k_5\boldsymbol{\xi}^T\boldsymbol{\xi} + (\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi})^T\boldsymbol{\xi} - \boldsymbol{\xi}^T\boldsymbol{\ddot{\eta}}^\alpha \tag{21}$$

If the coefficients satisfy the condition of $k_5 = 1/(4k_4)$, then Eq. (21) can be transformed to the following form:

$$\dot{V}_2 = -k_4(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi} - \boldsymbol{\xi})^T(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi} - \boldsymbol{\xi}) - \boldsymbol{\xi}^T\boldsymbol{\ddot{\eta}}^\alpha \tag{22}$$

Since $\boldsymbol{\xi}$ and $\boldsymbol{\ddot{\eta}}^\alpha$ are bounded, then $\left|\boldsymbol{\xi}\ \boldsymbol{\ddot{\eta}}\right| \le \Delta$ can be obtained, where Δ_1 is an arbitrary small positive constant. Then, Eq. (23) is obtained as:

$$\dot{V}_2 \le -k_4(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi} - \boldsymbol{\xi})^T(\boldsymbol{J}_s^T\boldsymbol{e}_h - \boldsymbol{\varphi} - \boldsymbol{\xi}) + \Delta_1 \tag{23}$$

Thus, V_2 is asymptotically converged, and the same applies to $(\boldsymbol{e}_h, \varphi, \xi)$.

Substituting Eq. (2) into $\boldsymbol{J\dot{v}} = -k_5\boldsymbol{\xi} - \boldsymbol{\dot{J}v}$, the initial form of control input $\boldsymbol{\tau}$ is given as $\boldsymbol{\tau}'$:

$$\boldsymbol{\tau}' = -k_5\boldsymbol{MJ}^T\boldsymbol{\xi} - \boldsymbol{MJ}^T\boldsymbol{\dot{J}v} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{Dv} \tag{24}$$

where $\boldsymbol{\tau}' = [\tau'_{u1}, \tau'_{v1}, 0, \tau'_{u2}, \tau'_{v2}, 0, \cdots, \tau'_{un}, \tau'_{vn}, 0]^T$.

In the preliminary design of control input, the yaw angle of each USV is not designed, and the motion control mainly relies on τ'_u and τ'_v. Under the input of Eq. (24), the yaw rate acceleration is $\dot{r}_i = 0$ for each USV. In other words, USVs will sail without making a turn. However, this drive configuration and motion pattern do not accord with the practical situation. Therefore, the control input needs to be reshaped considering the underactuated feature of USVs, which is called the control allocation.

CONTROL ALLOCATION

The control allocation can be achieved by properly designing the yaw angle, as shown in Fig. 3. By defining the yaw angle error and applying the sliding-mode control method, the yaw moment τ_r is designed. In the control allocation layer, the surge thrust τ_u remains unchanged, i.e. $\tau_u = \tau'_u$, while the sway thrust τ_v is set equal to zero, and the yaw angle varies in real time to eliminate the demand of τ_v.
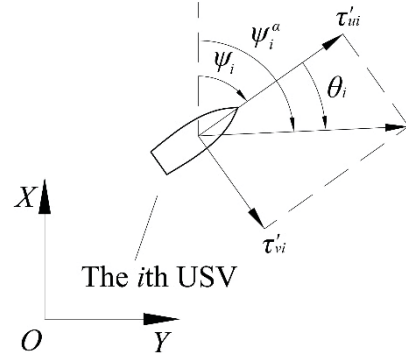

Fig. 3. Schematic of yaw angle designing

The yaw angle error of the ith USV is defined as

$$\tilde{\psi}_i = \psi_i - \psi_i^\alpha = -\theta_i \tag{25}$$

where ψ_i^α is the desired value of ψ_i, and $\theta_i = \text{atan}\,2(\tau'_{vi}, \tau'_{ui})$.

Then, in order to eliminate the yaw angle error, the following first-order sliding manifold is introduced:

$$S_i = \dot{\tilde{\psi}}_i + \lambda\tilde{\psi}_i = r_i - \dot{\psi}_i^\alpha - \lambda\theta_i \tag{26}$$

and the Lyapunov function V_3 is defined as

$$V_3 = \frac{1}{2}S_i^2 \tag{27}$$

Differentiating both sides of Eq. (27), and setting $\dot{r}_i$ as follows

$$\dot{r}_i = -\lambda r_i + (\lambda\dot{\psi}_i^\alpha + \ddot{\psi}_i^\alpha) - k_6\,\text{sgn}(S_i) \tag{28}$$

we get

$$\dot{V}_3 = S_i\dot{S}_i = S_i[\dot{r}_i + \lambda_2 r_i - (\lambda\dot{\psi}_i^\alpha + \ddot{\psi}_i^\alpha)] = -k_6\left|S\right| \le 0 \tag{29}$$

Thus, the sliding surface S_i is asymptotically stable, and the same applies to $\tilde{\psi}_i$. Substituting Eq. (1) into Eq. (28), the yaw moment of the ith USV is obtained as follows:

$$\tau_{ri} = -k_6 m_{33i}\,\text{sgn}(S_i) + (d_{ri} - \lambda m_{33i})r_i - \frac{m_{11i} - m_{22i}}{m_{33i}}u_i v_i + (\lambda\dot{\psi}_i^\alpha + \ddot{\psi}_i^\alpha) \tag{30}$$

To avoid differentiating of ψ_i^α, the following second order filter is introduced to calculate $\dot{\psi}_i^\alpha$ and $\ddot{\psi}_i^\alpha$.

$$\ddot{\psi}_i^\alpha + \zeta\dot{\psi}_i^\alpha + \omega(\psi_i^\alpha - \psi_i - \theta_i) = 0 \tag{31}$$

where ζ and ω are filter gains, which are to be properly chosen.

NUMERICAL SIMULATION

To illustrate that the proposed methodology is effective, numerical simulations of USV swarm motion have been performed. The swarm comprises 6 homogeneous USVs. The total length of each USV is 13 meters, and the motion parameters of USVs are given as follows [9]: m_{11}=120×10³ kg, m_{22}=177.9×10³ kg, m_{33}=636×10⁵ kg, d_u=215×10² kg·s⁻¹, d_v=147×10³ kg·s⁻¹, d_r=802×10⁴ kg·s⁻¹. In order to be consistent with the actual situation, the actuator saturation is taken into account and the saturation limits of control inputs for each USV are set as $|\tau_{ui}| \leq 1.0 \times 10^6$N and $|\tau_{ri}| \leq 1.0 \times 10^9$Nm.

The control parameters mentioned in the controller design are chosen as follows: k_1=20, k_2=5, k_3=0.1, k_4=0.5, k_6=2, λ=0.2, ζ=10, ω=20. While selecting parameters, some aspects need special attention. The collision avoidance task has higher priority than the tele-operated task and swarm dispersion, so it is advisable to set k_1 and k_2 relatively large, while k_3 only needs to be a small variable. Parameters k_4 and k_6 affect directly only the convergence rate of error variables, but if they are too large, the system response will also be affected. The distance parameters mentioned in the autonomous task have been chosen as follows: R_o=50m, r_o=25m, R_r=20m, r_r=10m.

SIMULATION 1: TARGET TRACKING WITH STATIC OBSTACLES IN THE VICINITY OF THE COURSE

In simulation 1, the USV swarm tracks a moving underwater target, and there are some static obstacles in the navigation area. The target position (x_t, y_t) is assumed always known, and the desired distance variance of the swarm is chosen equal to 80m², thus the desired position function is designed as X_h=[x_t, y_t, 80m²]. The target starts moving from point (40m, 180m) at time t=0s, and reaches points (250m, 450m), (150m, 650m), (-50m, 450m) at times t=30s, 55s, and 80s, respectively. Besides, since the target is under water, the USVs will not collide with the target. The initial velocities of all USVs are set equal to zero. The overall performance of the USV swarm is illustrated in Fig. 4 through Fig. 8.

Fig. 4 presents the sailing paths of USVs and moving target. The six colored lines denote the paths of six USVs, while the black dotted line denotes the target path. As we can see in the subfigures, the swarm has a good performance in target tracking, and each USV can automatically choose a proper route to avoid collisions when getting close to the obstacle. It is easily noticeable in Fig. 4(b) that when the obstacle is too close to the target, the six USVs choose to bypass the obstacle in different directions. Besides, during the navigation, each USV keeps a safe distance from other USVs, and collision between USVs does not happen.
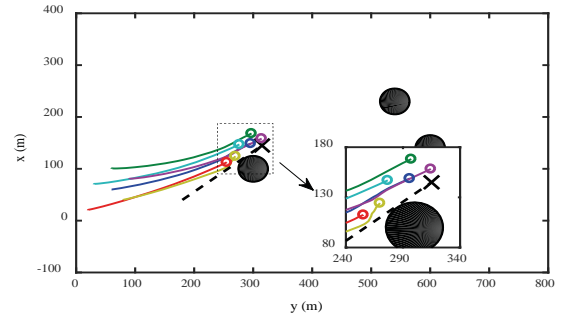
(a) Time t = 15s
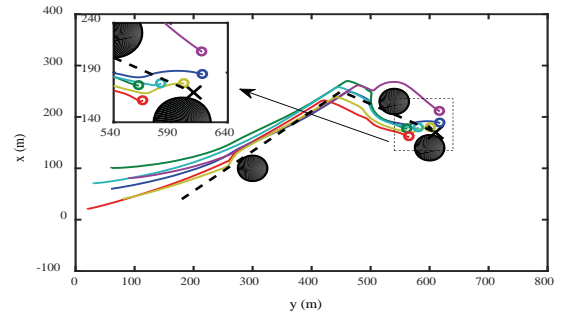
(b) Time t = 50s
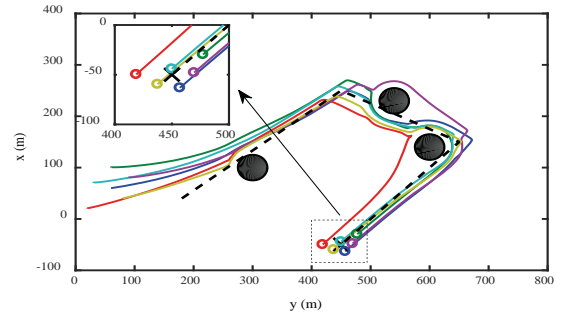
(c) Time t = 80s

Fig. 4. Sailing path of the USV swarm

Fig. 5 shows the error curves of the tele-operated task, where e_1 and e_2 denote the relative positions of the swarm center and the target, and e_3 denotes the error of desired distance variance. The most expected case is that e_1, e_2 and e_3 all converge to zero. However, in this simulation, due to the existence of obstacles, these error variables will not converge to zero until the swarm has sailed away from the obstacles. This phenomenon is explained by the fact that collision avoidance has higher priority than target tracking and formation keeping.

In order to see how each USV works, let us take USV 1 as an example. Fig. 6 shows the desired yaw angle error of USV 1, while Fig. 7 and Fig.8 show the input of thrust and yaw moment respectively. In Fig. 6, most of the time, the error of the desired yaw angle keeps equal to zero, while the large error values at times 30s and 55s are caused by surrounding obstacles. In Fig. 7 and Fig. 8, high frequency oscillations of control inputs are observed. These oscillations are due to the combined effect of tracking the desired position with virtual

repulsive forces from adjacent USVs and from the obstacle. After the obstacle avoidance is finished, the control inputs get stable. From the sailing paths shown in Fig. 4 we can find that these oscillations do not affect the performance of target tracking and obstacle avoidance. It is important to note that in practical applications, USVs may get a larger spacing than that assumed in the simulation, or smaller USVs may be used at the same spacing extent. In those situations the control inputs will be much smoother when adjusting position and orientation.
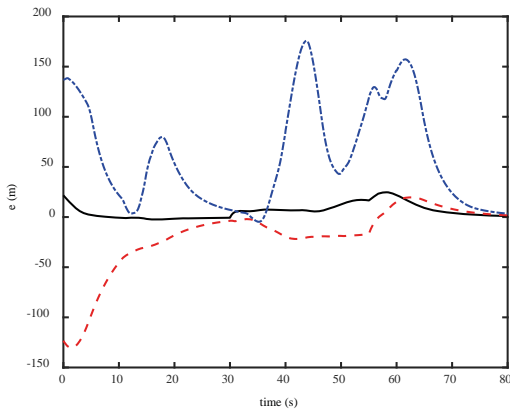
Fig. 5. Response curves of tele-operated task error
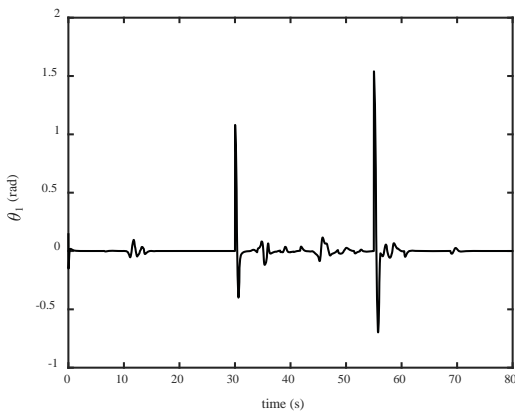
Fig. 6. Desired yaw angle error of USV 1
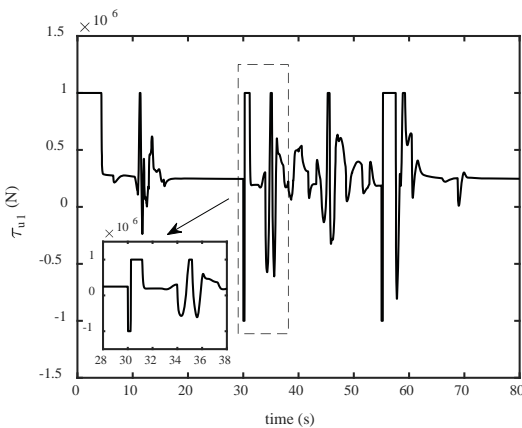
Fig. 7. Thrust input of USV 1
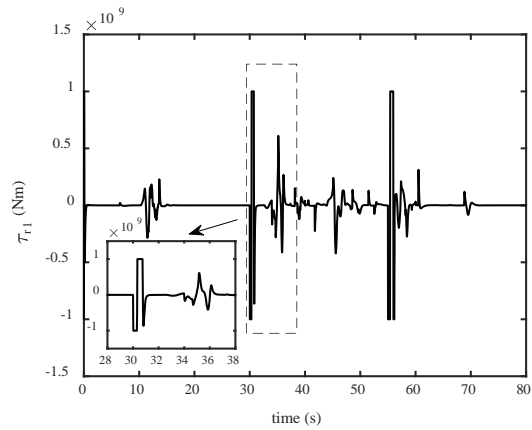
Fig. 8. Yaw moment input of USV 1

SIMULATION 2: TARGET TRACKING WITH MOVING OBSTACLES IN THE VICINITY OF THE COURSE

In simulation 2, the USV swarm tracks a moving underwater target, and there are three moving obstacles in the navigation area. The target position (x_t, y_t) is assumed always known, and the desired distance variance of the swarm is chosen equal to 80m², thus the desired position function is designed as X_h=[x_t, y_t, 80m²]. The target starts moving from point (40m, 180m) at time t=0s and moves at constant speed to reach points (250m, 500m), (440m, 590m), (585m, 725m) at times t=40s, 60s, and 80s, respectively. The initial velocities of all USVs are set equal to zero. The overall performance of the USV swarm is illustrated in Fig. 9 through Fig. 13.
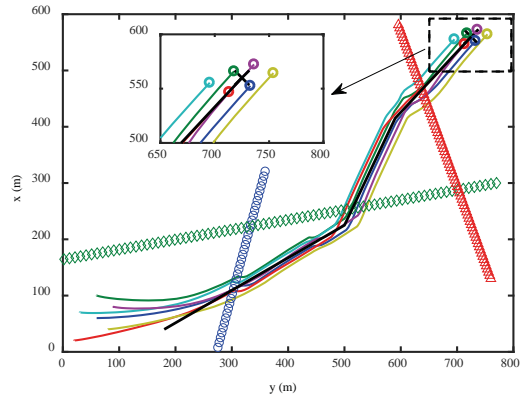
Fig. 9 shows the sailing path of USV swarm at times t=20s, 34s and 80s. In the subfigures, the black polyline denotes the path of the moving target, while the six colored curves denote the tracking paths of six USVs. As we can see in Fig. 9(a) and Fig. 9(b), there are 3 ships sailing in the vicinity of the swarm, and they are equivalent to moving obstacles for the USVs. It is easily noticeable in the enlarged drawing fragments that USVs automatically avoid the obstacle ships. In this simulation, each USV tracks the moving target well with acceptable overshoot at turning points. Initially, the 6 USVs are far apart from each other, then they gather rapidly and form a relatively stable formation shape while tracking the target. During the auto navigation, the 6 USVs always keep a safe distance from each other, and no collision between USVs occurs.
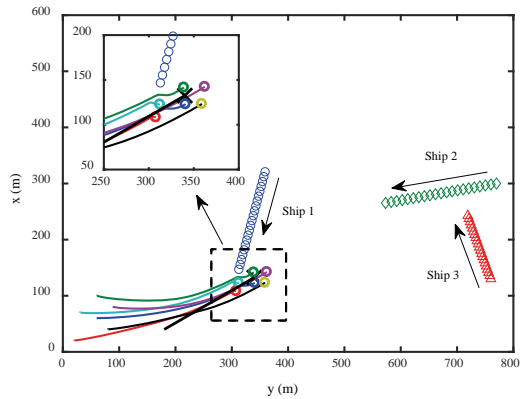
(a) Time t = 20s
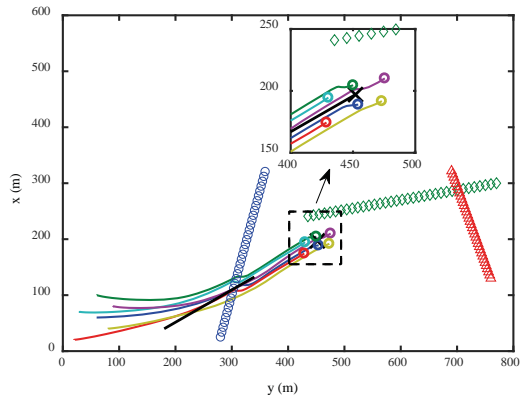
(b) Time t = 34s

(c) Time t = 80s

Fig. 9. Sailing path of the USV swarm

Fig. 10 shows the error curves of the tele-operated task, and Fig. 11 shows the desired yaw angle error of USV 1. In Fig. 10, it is easy to notice that the initial errors are rapidly eliminated, which means that the swarm reaches the desired position and gets the desired distance variance gradually. As for the rebounds at times t=17s, 40s, and 65s, they are caused by the presence of the obstacles, as a result of which the USVs need to take time for making a turning. The underlying reason has been explained in detail in simulation 1.

Fig. 12 and Fig. 13 show the thrust and yaw moment input of USV 1. Combining with Fig. 11, we can see that the

controller is running smoothly, except for the oscillations at times 17s, 40s, and 65s, generated while avoiding the sailing ships. Once the swarm sails away from the obstacle ships, the error variables converge to zero rapidly. We can conclude from the simulation results that the proposed USV swarm control method has a good performance in target tracking and obstacle avoidance.
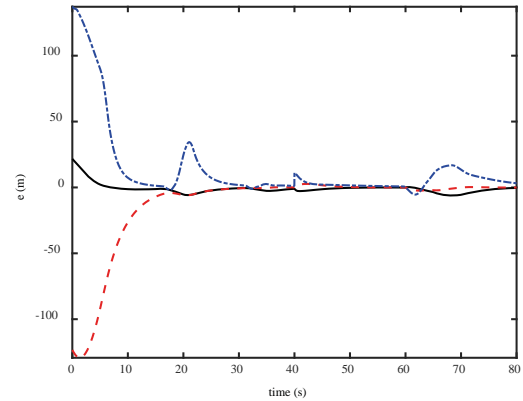

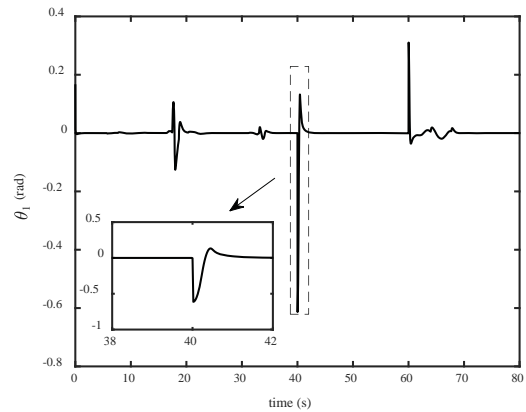*Fig. 10. Response curves of tele-operated task error*

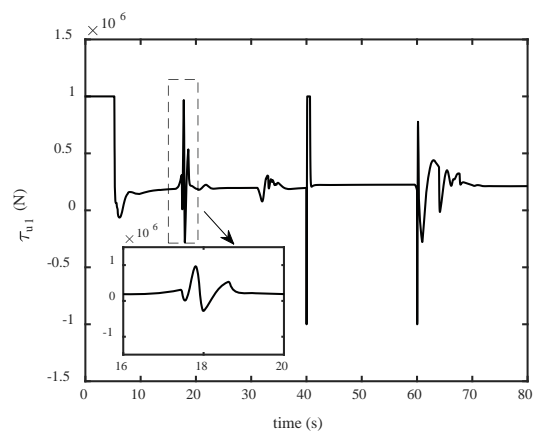*Fig. 11. Desired yaw angle error of USV 1*

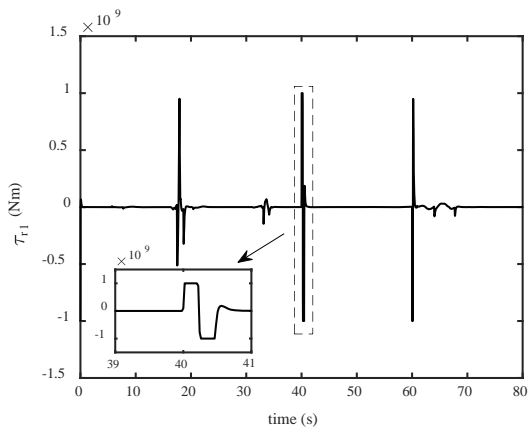*Fig. 12. Thrust input of USV 1*

Fig. 13. Yaw moment input of USV 1

CONCLUSIONS

In the paper, a hierarchical control framework with relevant algorithms is proposed to achieve auto navigation of a USV swarm. This control enables the swarm to track autonomously the desired target, avoid obstacles, and avoid collisions between swarm members. Theoretical proof and numerical simulations are provided to demonstrate the efficiency and robustness of the overall control system.

It can be seen from the simulation results that the originally designed hierarchical swarm controller based on potential field method and individual control allocation design is suitable for the underactuated USV swarm. Besides, compared to other formation control methods, a fixed formation shape is not needed in the swarm control, and the swarm members are intelligent to a certain extent.

In the future study, the proposed USV swarm control method will be improved to facilitate its conversion into practical use. For example, an actual electronic chart will be pixelated and used for modeling external geographical information for USVs.

REFERENCES

1. Do K. D. Practical formation control of multiple underactuated ships with limited sensing ranges. Robotics and Autonomous Systems, 2011, 59(6): 457-471.

2. Zhuang J.Y., Su Y.M., Liao Y.L., Sun H. Unmanned surface vehicle local path planning based on marine radar. Journal of Shanghai Jiaotong University, 2012, 9: 006.

3. Shojaei K. Neural adaptive robust control of underactuated marine surface vehicles with input saturation. Applied Ocean Research, 2015, 53:267-279.

4. Wai R.J., Liu C.M., Lin Y.W. Design of switching path-planning control for obstacle avoidance of mobile robot. Journal of the Franklin Institute, 2011, 348(4): 718-737.

5. Cui R., Ge S.S., How B.V.E., Choo Y.S. Leader–follower formation control of underactuated autonomous underwater vehicles. Ocean Engineering, 2010, 37(17): 1491-1502.

6. Peng Z., Wang D., Chen Z., Hu X. Adaptive dynamic surface control for formations of autonomous surface vehicles with uncertain dynamics. IEEE Transactions on Control Systems Technology, 2013, 21(2): 513-520.

7. Ding L., Guo G. Formation control for ship fleet based on backstepping. Control and Decision, 2012, 27(2): 299-303.

8. Mehrjerdi H., Ghomman J., Saad M. Nonlinear coordination control for a group of mobile robots using a virtual structure. Mechatronics, 2011, 21(7): 1147-1155.

9. Do K.D. Formation control of underactuated ships with elliptical shape approximation and limited communication ranges. Automatica, 2012, 48: 1380-1388.

10. Glotzbach T., Schneider M., Otto P. Cooperative line of sight target tracking for heterogenous unmanned marine vehicle teams: From theory to practice. Robotics and Autonomous Systems, 2015, 67: 53-60.

11. Johnson J.T. A Brief Investigation of Swarm Theory and Applications. ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, 2009: 209-218.

12. Bae J., Kim Y. Adaptive controller design for spacecraft formation flying using sliding mode controller and neural networks. Journal of the Franklin Institute, 2012, 349(2): 578-603.

13. Wang H. Flocking of networked uncertain Euler–Lagrange systems on directed graphs. Automatica, 2013, 49(9): 2774-2779.

14. Cepeda-Gomez R., Olgac N., Sierra D.A. Application of sliding mode control to swarms under conflict. IET control theory & applications, 2011, 5(10): 1167-1175.

15. Franchi A., Secchi C., Son H.I., Bülthoff H.H. Bilateral teleoperation of groups of mobile robots with time-varying topology. IEEE Transactions on Robotics, 2012, 28(5): 1019-1033.

16. Liu Y.C. Task-space coordination control of bilateral human–swarm systems. Journal of the Franklin Institute, 2015, 352(1): 311-331.

17. Wan L., Dong Z.P., Li Y.M., He B. Global asymptotic stabilization control of incomplete symmetry underactuated USV. Journal of Huazhong University of Science and Technology, 2014, 8(42): 48-53.

18. Liu Y.C., Chopra N. Controlled synchronization of heterogeneous robotic manipulators in the task space. IEEE Transactions on Robotics, 2012, 28(1): 268-275.

19. Stipanović D.M., Hokayem P.F., Spong M.W., Siljak D. Cooperative avoidance control for multiagent systems. Journal of Dynamic Systems, Measurement, and Control, 2007, 129(5): 699-707.

20. Schwager M., Rus D., Slotine J.J. Decentralized, adaptive coverage control for networked robots. The International Journal of Robotics Research, 2009, 28(3): 357-375.

CONTACT WITH THE AUTHORS

Lei Zhang
e-mail: cheung103@163.com

Harbin Engineering University
Nantong Street
150001 Harbin
China