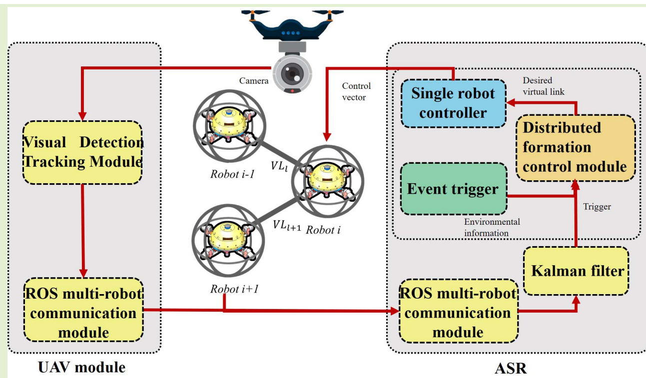
A Virtual Linkage-Based Dual Event-Triggered Formation Control Strategy for Multiple Amphibious Spherical Robots in Constrained Space With Limited Communication

He Yin, *Student Member, IEEE*, Shuxiang Guo, *Fellow, IEEE*, and Meng Liu, *Student Member, IEEE*

***Abstract*—Aiming to deal with the formation maintaining in constrained space for a group of robots, this paper proposed a virtual linkage-based dual event-triggered formation control strategy. Based on the idea of virtual linkage and event-trigger, a dual event trigger is proposed to guarantees robot formation keeping and collision avoidance. Unlike classical event-triggered control, the control strategy triggered by the detection of the state of virtual linkage determine the motion mode of the robot, which can reduce the control updating and eliminate continuous communication between robots. Moreover, an adaptive control law inspired by the motion of a simple pendulum was developed to adjust the given angle of the virtual linkage in the case that the multi-robot system pass the restricted region. Finally, to validate the performance of the formation control strategy, simulations and a series of experiments in an indoor swimming pool are presented. The results demonstrate the robustness and feasibility of the proposed formation control strategy.**



***Index Terms*—Formation control, event-triggered control, collision avoidance, limited communication, multiple amphibious spherical robots.**

I. Introduction

With the development of littoral environment more and more frequently, the demand for littoral exploration technology and equipment is increasing. Autonomous amphibious robots are becoming increasingly popular in marine engineering [1], [2]. With the increasing complexity of tasks, it is necessary to carry out underwater missions which are beyond the capacity of a single autonomous underwater vehicle(AUV). In order to explore the ocean more efficiently, the cooperative mechanism and strategy are used to control multiple underwater autonomous robots to accomplish tasks in a cooperative manner [3], [4].

Formation control is an important research area of cooperative control of multiple robot. Compared with the single robot, the overall performance and efficiency of multi robots that form a certain geometric shape are improved. The existing formation control approaches can be classified into three categories: the leader-follow method, the swarm behavior method and the virtual structure method. In the leader-follow method [5], [6], some robots are regarded as the leader, and the other robots are regarded as the follower. The follower robots need to maintain a certain angle and distance with the leader robots. In the swarm behavior method [7], [8], each robot has a series of basic designed behaviors. Each robot determines its own behavior according to the environment and the behavior of its neighbors. The virtual structure method considers the multi-robot system as a virtual single rigid body and each robot is considered as a virtual part of the rigid body [9]. This method realizes the formation control by changing the state of the rigid body. In practical applications, in order to allow multiple robots to pass the restricted areas, the transformation

Manuscript received February 28, 2022; revised May 10, 2022; accepted May 10, 2022. Date of publication May 17, 2022; date of current version July 1, 2022. The associate editor coordinating the review of this article and approving it for publication was Prof. Kazuaki Sawada. *(Corresponding author: Shuxiang Guo.)*

He Yin and Meng Liu are with the School of Life Science, Beijing Institute of Technology, Beijing 100081, China (e-mail: yinhe@bit.edu.cn; liumeng@bit.edu.cn).

Shuxiang Guo is with the Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, Ministry of Industry and Information Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: guoshuxiang@bit.edu.cn).

Digital Object Identifier 10.1109/JSEN.2022.3175715

1558-1748 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

of formation in complex environments has gradually attracted people's attention. In [10], An adaptive self-organizing map neural network method is proposed for distributed formation control, in which, the group of AUVs change its shape as needed when they are approaching to the desired locations. In [11], the multi robot system is considered as a virtual linkage which consists of different virtual links. The formation shape of multi robot system can be changed by controlling the state of the virtual links. In [12], the role switching event trigger and a formation scaling factor are introduced to scale up or scale down the given formation size during the formation tracking. In [13], a self-organized formation strategy is proposed to realize the adaptive collective organization and dynamic circle formation.

In many researches on multi-robot collaborative control, it is assumed that robots can access to continuous measurements and control signals. However, in practical applications, the traditional continuous control is difficult to be applied to robots due to the limitation of communication and computation resources. Different from the traditional continuous control, the control signal of the event-triggered control strategy is discrete. Specifically, the control signal update and communication of each robot only occur when event triggering conditions are satisfied. In other words, event-triggered theory only sends and receives really "necessary" state signals to achieve the same control effects as the continuous control, which requires less computational and communication loads. Since less information is received, event trigger conditions and controllers are the key problem to the design of event-triggered control algorithms. Thanks to this advantage, event-triggered cooperative control has been studied subsequently [14]–[18]. In [19], a dynamic event-triggered communication mechanism and an event-triggered formation protocol was designed to achieve the better formation performance by using only locally triggered sampled data. In [20], a consensus protocol and an ETC strategy based on a closed-loop state estimator is proposed and it is verified by applying it to multi-vehicle platoon control in simulations. In addition, there are many researches on event-triggered control in practical applications, such as times delay and packet loss [21]–[23]. However, there are few results in which event-triggered algorithms are applied to practical multi-robot systems. It is also worth noting that formation protocol designed in those papers mentioned above are presented in the open space. But the working environment of robots is often complex, especially the underwater environment. The challenges we face are as follows: 1) How to combine obstacle avoidance control strategy and event-triggered control to achieve obstacle avoidance during formation keeping with limited information. 2) How to design an event-triggered control strategy for a multi-robot system and integrate it into a fully autonomous multi-robot system with hardware and software release. Therefore, the researches on incorporating collision avoidance method into event-triggered formation control is studied in some related literatures, such as [24] and [25], in which, collision avoidance method was introduced as an auxiliary variable in the event-triggered formation protocol, such that robots can balance formation tracking and collision avoidance. In [26], a safe navigation method combined vision-based path planning with multivariable event-triggered controller is proposed to minimize the controller effort. Comparative simulation and experiments on mobile robot are done to verify robustness of the proposed controller. To solve the problem of flight stability at the condition of obstacles and limited thrust, a parallel-triggered scheme is proposed in [27], in which collision avoidance method and event-triggered control are introduced to the planning layer to determine the update moment of the controller.

Currently, the traditional continuous control relies on the continuous measurements and control signals to achieve the formation control. Although the event-triggered formation control can save the limited network resources in the multi-robot system, the obstacle avoidance problem in practical applications is rarely considered. Therefore, this paper proposed a virtual linkage-based dual event-triggered formation control strategy. Through the introduction of dual event-triggered, the multi-robot system requires less computational and communication loads in the case that the multiple amphibious spherical robots(ASR) system pass the restricted region. The main contribution of this paper are as follows:

1) The improved virtual linkage formation strategy and edge-based event-triggered algorithm were combined to establish the event-trigger algorithm that uses the virtual link state as the event trigger condition.
2) In design of the event trigger, mode triggers and communication triggers are introduced to allows the multi-robot system reduce the control updating and eliminate continuous communication during passing the restricted region.
3) An adaptive control law inspired by the motion of the simple pendulum was developed to adjust the given angle of the virtual linkage in the case that the multi-robot system pass the restricted region.

The rest of this paper is organized as follows. Section II presents a brief introduction about our new generation amphibious spherical robot and the multi-ASR system. The formation control strategy for multiple amphibious spherical robots is presented in Section III. Simulation and results are provided in Section IV to assess the performance of the proposed approach. The effectiveness and robustness of the proposed formation strategy are verified through experiments in Section V. Section VI provides a conclusion of the whole paper.

II. The System Overview of the Multiple Amphibious Spherical Robots

A. The Structure of Amphibious Spherical Robot

On the basis of the robot mentioned in the references [28]–[31], a new generation of amphibious spherical robot is developed with improved performance. The corresponding specifications of the robot are provided in TABLE I. Its mechanical structure is inspired by the morphology and locomotion modes of turtle, whose conceptual design and prototype is shown in Fig. 1. The shape of the robot is designed as a ball to reduce the resistance when the robot moving in the water. The structure of the robot is composed of upper

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

TABLE I
TECHNICAL PARAMETERS OF THE ROBOT

Items	Characteristics
Size(L×W×H)	300mm×300mm×300mm
Total mass	6.7kg
Number of body joints	12
Drive mode	DC motors, digital servomotors
On-board sensors	Binocular cameras, pressure sensors, IMU
Power supply	DC 8V
Operation time	1.5h
Controller	STM32F407, NVIDIA Jetson TK1
Maximum crawling speed	6.05cm/s
Maximum swimming speed	60 cm/s
Maximum turning rate	16.80°/s

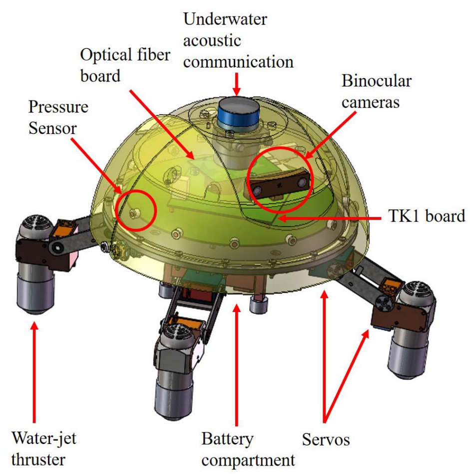


Fig. 1. The conceptual design of proposed spherical robot amphibious spherical robot.

Fig. 2. The robotic prototype of the multi-ASR system.

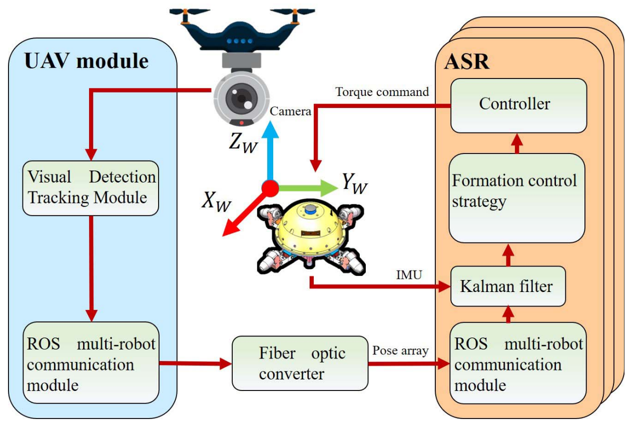


Fig. 3. The structure of the multi-ASR system.

and lower parts. The upper part of the robot contains various sensors and control modules, which perceive environmental information and control the movement of the robot. The motion mechanisms is mainly located in the lower part of the robot, which is composed of water jet thrusters and waterproof servo motors. The water jet thrusters can push the robot to swim in the water, and waterproof servo motors can help the robot crawl like a tortoise on land. The maximum speed of this amphibious robot in the water can reach 60 cm/s, and the maximum crawling speed of the robot on land is 6.05cm/s. In addition, a battery located at the bottom of the robot is used to power the robot.

The robot is equipped with several sensors and controllers to receive environmental information and control commands from the operator. The NVIDIA Jetson TK1 located in the upper part of the robot is the main controller of the robot. Its functions include receiving external instructions, analyzing environmental information, and controlling robot movement, etc. The STM32 is the bottom controller of the robot, it can directly control the mechanical legs according to the commands from the main controller. The robot is also equipped with an inertial measurement unit (IMU) to get the posture information of the robot. The depth information of the robot in the water can be measured by the pressure sensor. In addition, the surrounding environment images of the robot can be obtained by binocular cameras.In our experiment, we also used a host PC to monitor the status information of all robots with the help of remote desktop software.

B. Principle of Multi-ASR System

The multi-ASR system can accomplish many complex tasks through robotic collaboration. The ASR based multi-robot platform as shown in Fig. 2 has been developed for exploration of cooperative control strategies. The structure of the multi-ASR system is shown in Fig. 3, which contains two amphibious spherical robots and a unmanned aerial vehicle(UAV) module. The camera equipped on the UAV module can provide position information for the ASR. The Kalman filter is used to predict the position of the robot. The robots can communicate with each other through ethernet with ROS multi-robot communication module..

In our experiments, the four legs of each robot were set in the X shape. This way of moving underwater can overcome current disturbances created by other robots and ensure precise position control. Since the force bearing point is located in the center of the X-shaped robot, the torque generated by the mechanical legs is small for the ASR, which will lead to difficulties of ASR heading control. What's more, it is difficult for robots to adjust the vision of binocular cameras by adjusting

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

their heading. In addition, the processing of binocular camera image data consumes a lot of computing resources, which will slow down the corresponding speed of the robot. To solve the above problem, we emulate binocular visual information by global visual information conversion. By doing so, we can test distributed cooperative control algorithms that involve only local neighbor-to-neighbor information exchange due to limited communication or sensing.

III. The Formation Control Strategy of Multiple Amphibious Spherical Robots

In this section, we design an edge-based dual-event trigger strategy and a virtual linkage-based adaptive formation control strategy for ASR, which can comply with the following four requirements: 1) Reduce the communication and driving frequency of the robot; 2) collision avoidance; 3) maintain a desired geometric pattern; 4) adaptive formation transformation. In the previous section, from the perspective of the single amphibious spherical robot and the multi-ASR system, the design requirements have been basically satisfied. The principles and methods will be analyzed in this section, and the details of this section will be introduced from the following aspects, the problem statement, the formation control architecture, the improved virtual linkage algorithm, the design of dual event trigger, and the adaptive formation strategy.

A. Problem Statement

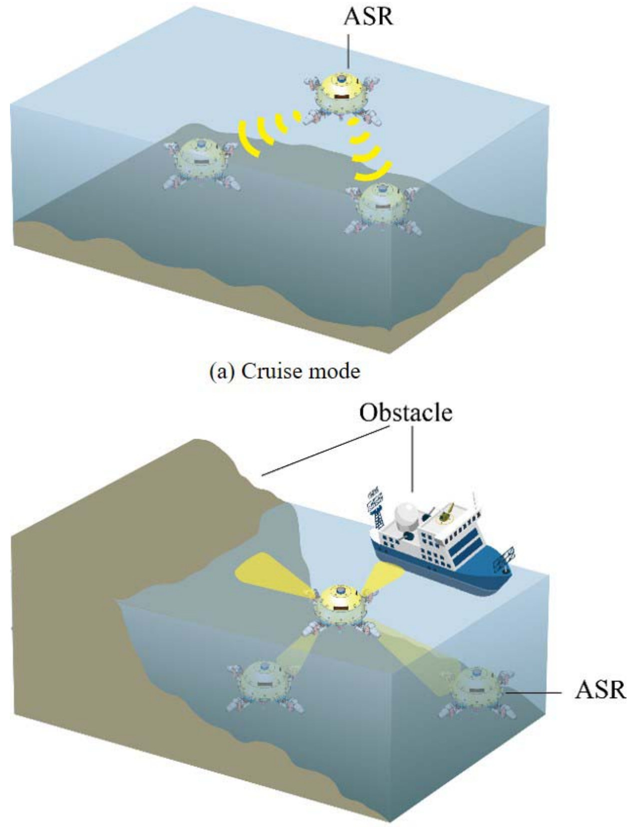
In the multi-amphibious spherical robot system, each robot with the similar dynamics described by the following linear equation

$$\dot{\boldsymbol{p}}_i(t) = \boldsymbol{u}_i(t), \quad i = 1, 2, \ldots, n, \tag{1}$$

where $\boldsymbol{p}_i(t) = [x_i(t), y_i(t), z_i(t)]^T \in \mathbf{R}^3$ is the position of robot i in the inertial frame; $\boldsymbol{u}_i(t) = \left[x_i^u(t), y_i^u(t), z_i^u(t)\right]^T \in \mathbf{R}^3$ represents the control vector of robot i. The $x_i^u(t)$, $y_i^u(t)$ and $z_i^u(t)$ represent component of the control vector in the body-fixed frame at time t.

The information exchange in multiple amphibious spherical robot systems containing N robots can be represented by a directed graph network $G = \{V, E\}$, where the vertex set $V \triangleq \{v_i \mid i = 1, 2, \ldots, N\}$ represents the set of nodes in the graph, and $E \triangleq \left\{\boldsymbol{e}_{ij} \mid \boldsymbol{e}_{ij} \triangleq \left(v_i, v_j\right), i = 1, 2, \ldots, N, j \in \mathcal{N}_i\right\}$ is the set of directed edges. $\mathcal{N}_i$ is the neighbor set of robot i, which can be defined by $\mathcal{N}_i \triangleq \left\{v_i \in V \mid \left(v_i, v_j\right) \in E, i \neq j\right\}$. An edge $\boldsymbol{e}_{ij}$ in the digraph G denotes that robot j can obtain information from robot i. Then i is the parent node, and j is the child node. We assume that the directed graph network G has M directed edges, the E can be also defined as $E \triangleq \left\{\boldsymbol{e}_l \mid \boldsymbol{e}_l = \boldsymbol{p}_i(t) - \boldsymbol{p}_j(t), l = 1, 2, \ldots, M\right\}$, which denotes that the distance between two robots. A directed spanning tree of a digraph is the node in the directed tree can nodes of the graph. In particular, assumes that the directed graph G contains a spanning tree.

Let $A = \left[a_{ij}\right]_{N\times N}$ be the adjacency matrix of the directed graph G, where $a_{ij} = 1$, if $\boldsymbol{e}_{ij} \in E$, otherwise $a_{ij} = 0$. Since each edge $\boldsymbol{e}_{ij}$ has the direction, A is an unsymmetrical matrix. Let $D = [d_{il}]_{N\times M}$ be the incidence matrix of the directed graph G, where $d_{il} = -1$, if robot i is the parent node of edge l; $d_{il} = 1$, if robot i is the child node of edge l, otherwise $d_{il} = 0$. Then let $\bar{D} = \left[\overline{d_{il}}\right]_{N\times M}$ be the in-incidence matrix, where $\overline{d_{ll}} = 1$, if $d_{il} = 1, d_{il} \in D$, otherwise $\overline{d_{il}} = 0$. Then the Laplacian matrix of G can be defined as $L = \bar{D}D^T$.



(a) Cruise mode

(b) Avoidance mode

Fig. 4. Two motion modes of the proposed formation control strategy.

In order to allow the robots in the multi-robot system to converge to the desired state, the goal of the formation control strategy can be defined as

$$\lim_{t\to\infty} \boldsymbol{e}_l(t) - \boldsymbol{e}_l^d(t) = 0, \tag{2}$$

where $\boldsymbol{e}_l^d(\mathrm{t}) = \boldsymbol{p}_i^d(t) - \boldsymbol{p}_j^d(t)$ is the state of the desired edge. $\boldsymbol{p}_i^d(t)$ and $\boldsymbol{p}_j^d(t)$ are the desired positions of robots i and j, respectively.

B. Distributed Formation Control Architecture

The proposed formation control strategy provides two motion modes for ASR, and the robot can switch the corresponding motion mode according to the environment information. In the open space, the robot will move in cruise mode, which can avoid continuous communication and control; in the restricted space, the robot will move in avoidance mode, which can access to continuous measurements and control signals. Fig. 4 shows an illustrative example of the proposed two motion modes with a formation composed of three ASRs. In the first case [see Fig. 4(a)], three amphibious spherical robots navigate in the open space. To prolongs the endurance

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

time, all robots are in cruise mode. In this motion mode, the robot's control signals and communication are discrete. Only when the specific conditions are satisfied, the control signal and communication of each robot will be updated to maintain the basic formation. In the second case [see Fig. 4(b)], three amphibious spherical robots navigate in obstacle environments, where boat and shore are obstacles. In order to maintain a desired geometric pattern and avoid obstacles, all robots are in avoidance mode. In this motion mode, the control signals and communication of the robots are continuous, which can ensure the safety of multi-robot systems in restricted areas.

Definition 1 (Cruise Mode): In a multi-robot system, the control signal of the robot in the cruise mode is discrete, and its movement is only affected by the desired state and neighboring robots.

Definition 2 (Avoidance Mode): In a multi-robot system, the control signals and measurements of the robot in avoidance mode are implemented in a continuous manner, and the robot's motion will be affected by neighboring robots and the environment.

The structure of the proposed distributed formation control strategy is shown in Fig. 5. The formation strategy consists of three parts: event triggers, distributed formation control module and single robot controller. Event triggers have two functions. One is to determine the motion mode of the robot. The other is to determine the trigger time of the robot in cruise mode. The event trigger selects the motion mode for the robot based on the environment information. Formation control algorithms are used to maintain formation and avoid obstacles in different motion modes. The single robot controller generates the corresponding control vector for the robot according to the desire state generated by the distributed formation control algorithm.

The basic idea of the whole proposed formation algorithm is as follows. In a multi-robot system, the mode event trigger measures environmental information in a continuous manner and determines the robot's motion mode based on the environmental information. Next, if the robot is in cruise mode, the communication event trigger determines whether to generate an event based on the status information of the neighbor robot. When a communication event is triggered, the formation control strategy generates a new control vector based on the environmental information, otherwise the control vector remains unchanged. If the robot is in avoidance mode, the robot will receive real-time environmental information and generate corresponding control vectors according to the formation control algorithm. Finally, the single robot controller receives the control vector generated by the formation control strategy and drives the robot.

C. The Improved Virtual Linkage Algorithm

The formation control algorithm is a part of the proposed formation control strategy, which can drive a group of robots to realize and preserve a desired geometric pattern. The virtual linkage algorithm is a formation control strategy proposed by Liu *et al.* [11]. Although it is an efficient formation control strategy suitable for multi-robots, it is difficult to apply it to the

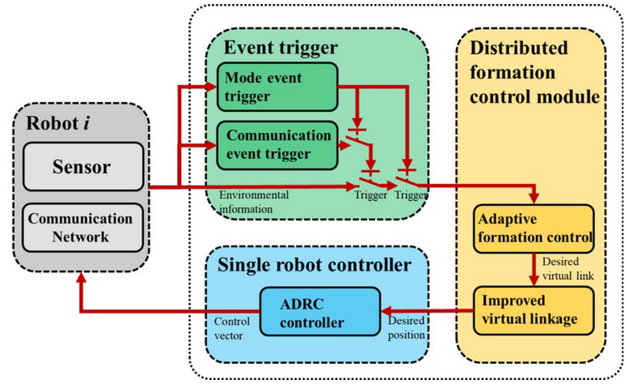


Fig. 5. The distributed formation control architecture.

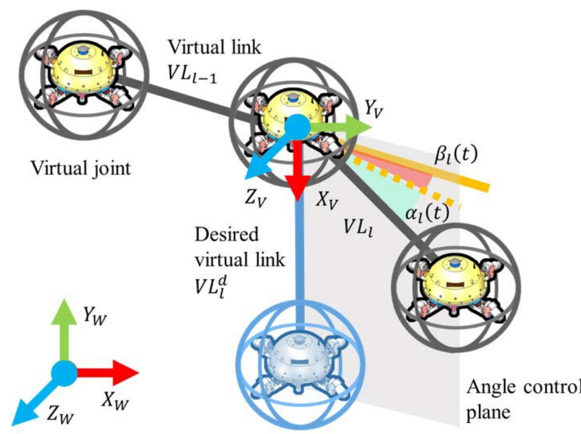


Fig. 6. The schematic diagram of a virtual linkage mechanism.

multi-ASR system in the three-dimensional underwater environment. In the proposed formation control module, we extend virtual linkage algorithm to propose a three-dimensional formation control strategy for multi-ASR system.

The virtual linkage is composed of two parts: virtual joints and virtual links. A virtual joint is a structure that connects two virtual links, which can limit the movement of the virtual links. The virtual link is a single rigid body, which can be connected by virtual joints. In the virtual linkage formation control strategy, a group of robots is considered as a virtual linkage, each robot is regarded as a virtual joint, and the edge formed by two robots is regarded as a virtual link. By coordinating the angles between each virtual link, the virtual linkage formation control strategy can let the robots in the multi-robot system converge to the desired state.

Consider that there are N robots and M edges in a multi-robot formation. Then this group of robots can be represented by N virtual joints and M virtual links. The principle of the virtual linkage is illustrated in Fig. 6. $\{O_W - X_W Y_W Z_W\}$ is the world coordinate system; $\{O_V - X_V Y_V Z_V\}$ is the virtual link coordinate system. In the virtual linkage, each robot is a virtual joint, VL_l is the virtual link, and VL_l^d is the desired virtual link. Similar to the principle of simple pendulum motion, the axis X_V of the virtual link coordinate system is regarded as the direction of gravity, which is parallel to VL_l^d.

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

In this way, the state of the virtual link VL_l will converge to the desired state VL_l^d under the control of the virtual linkage algorithm, which is similar to the movement of a simple pendulum affected by gravity. The Y_V axis of the virtual link coordinate system is perpendicular to VL_l^d and lies in the angle control plane formed by VL_l and VL_l^d. The Z_V axis of the virtual link coordinate system can be determined according to the right-hand rule. The angle between the virtual link VL_{l-1}^d and the angle control plane is denoted as $\beta_l(t)$. The angle between the virtual link VL_l and the projection of the extension line of the virtual link VL_{l-1}^d on the angle control plane is denoted as $\alpha_l(t)$. The state of the virtual link is defined as

$$\boldsymbol{L}_l(t)=\left[{}^{V}\boldsymbol{p}_l(t), \alpha_l(t), \beta_l(t), d_l(t)\right]^T, \quad l=1,2,\ldots,M, \tag{3}$$

where ${}^{V}\boldsymbol{p}_l(t) = [x_l(t), y_l(t), z_l(t)]$ is the position of the virtual joint in the virtual link frame; $d_l(t)$ is the length of the virtual link. Then the coordinates of the robot i in the world coordinate frame can be obtained by the following equation

$$\boldsymbol{p}_l(t) = {}^{W}T_V{}^{V}\boldsymbol{p}_l(t), \tag{4}$$

where

$${}^{W}T_V = \begin{bmatrix} R & \boldsymbol{p}_{l-1}(t)^T \\ 0 & 1 \end{bmatrix} \tag{5}$$

is homogeneous transform. The R can be written by the following equation (6), as shown at the bottom of the next page, $\sum\alpha_l = \sum_{i=1}^{l}\alpha_i(t)$ and $\sum\beta_l = \sum_{i=1}^{l}\beta_i(t)$ is the angular state of virtual link in the world frame.

D. The Design of Dual Event Trigger

The edge-based event-triggered algorithm is the formation control algorithm proposed by Wei *et al.* [32]. The basic idea is that only when the states of two adjacent robots meet the designed trigger conditions, the robot's state information and control signals can be updated.The proposed dual event trigger extends the previous edge-based event-triggered strategy from the following two aspects: combined the virtual linkage formation strategy and edge-based event-triggered algorithm, introduced mode triggers and communication triggers to realize obstacle avoidance. These improvements not only simplify the information transmitted between robots, but also allows the multi-robot system to maintain formation and avoid obstacles with limited communication and driving frequencies.

The dual event trigger is a part of the proposed formation control strategy, which can generate trigger moment to reduce the communication and driving frequency of the robot. The designed event trigger includes two event triggers: mode trigger and communication trigger. The priority of mode trigger is higher than the communication trigger. Only when the mode trigger selects the cruise mode, the communication trigger can generate trigger moment.

The principle of the mode trigger is very similar to the multi-instant gain-scheduling scheme. In the multi-instant gain-scheduling scheme, it provides many different controllers and a switching mechanism [33], [34]. Under this switching mechanism, the robot can select an appropriate controller according to the environment and its own state. Similar to the multi-instant gain-scheduling scheme, the proposed formation control strategy also has two motion modes. In different motion modes, the control laws of the robot are also different. And the mode trigger can determine the movement mode of the robot according to the environment information or the state of other robots. When the distance between the robot and obstacles or other robots is less than the warning distance, the mode trigger will select the avoidance mode, otherwise the mode trigger will select the cruise mode.

Event generation for mode triggers is affected by monitoring states and thresholds. The monitoring status based on artificial potential field can be written as

$$\begin{aligned} \boldsymbol{F}_i^o(t) &= \sum_{c=1}^{C}\boldsymbol{F}_{ic}(t) \\ &= \begin{cases} \displaystyle\sum_{c=1}^{C}\eta\frac{1}{(d_{ic}-d_0)^2}\frac{\boldsymbol{p}_i(t)-\boldsymbol{p}_c(t)}{d_{ic}}, & d_{ic}\le d_r \\ 0, & d_{ic}\ge d_r \end{cases} \end{aligned} \tag{7}$$

where η is the collision avoidance factor; r is the radius of the robot; d_{ic} is the shortest distance from the robot i to obstacle or the other robot k; d_0 is a small constant regarded as the safety distance to avoid collisions; d_r is the protection distance. when the distance d_{ic} from the robot to the obstacle or the other robot is close to protection distance, $\boldsymbol{F}_i^o(t)$ approaches 0, which can make the robot move more smoothly. When the distance d_{ic} from the robot to other robot or obstacle decreases, $\boldsymbol{F}_i^o(t)$ approaches infinity, which can make the robot avoid collisions. C is the total number of obstacles or other robots within radius d_{sen} of robot i. The neighborhood of the ith robot is defined using the standard Euclidean distance as $S_i = \{j : | \|x_i - x_j\| < d_{sen}\}$. The reason for choosing the artificial potential field(APF) as monitoring state is that it povides simple and effective solutions for practical application. In APF, each obstacle creates a potential field that pushes the robot away from them.

In the mode trigger, the trigger condition is defined as

$${}^{M}f_i\left(t, \boldsymbol{F}_i^o(t)\right) = \left|\boldsymbol{F}_i^o(t)\right| - {}^{M}\delta_i \tag{8}$$

where ${}^{M}\delta_i$ is the threshold of trigger condition.It reflects the detection of obstacles and the potential of collisions. Then the trigger moment of the avoidance mode is defined as

$${}^{M}t_{k+1}^i = \inf\left\{t >{}^{M} t_k^i \,|^{M}\ f_i\left(t, {}^{M}\delta_i\right) \ge 0\right\}, \tag{9}$$

The triggering condition (8) is monitored by each robot continuously. If ${}^{M}f_i\,(t, \boldsymbol{F}_i) < 0$, the mode trigger will select the cruise mode associated with rotbot i is satisfied at time $t = {}^{M}t_k^i$. The communication and control updates of the robot i are controlled by communication triggers, otherwise the mode trigger will select the avoidance mode and robot can access to continuous measurements and control signals. In different modes, the formation control strategy will generate different control vectors to control the robot.

The proposed mode trigger for robot i is shown in Fig. 7 The ordinate is the monitoring states of the mode trigger, which is affected by the distance between robot i and the

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

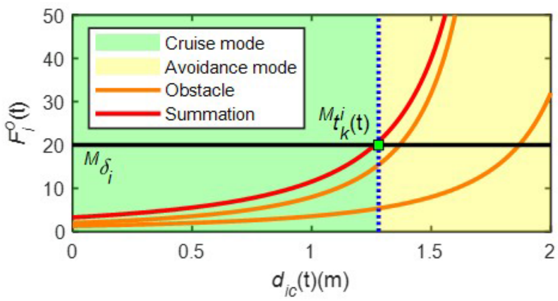
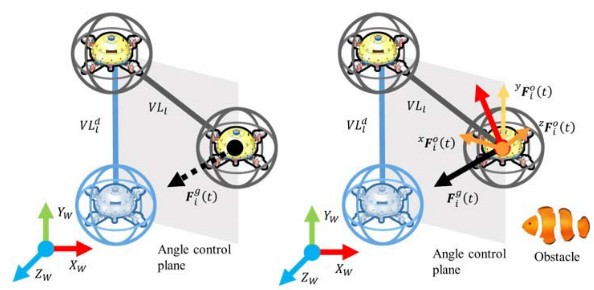


Fig. 7. The schematic of the mode trigger.



(a) Virtual link angle change in cruise mode.

(b) Virtual link angle change in avoidance mode.

Fig. 8. Two examples of virtual link angle change in different mode.

obstacle. The two orange curves in the figure are the potential fields of two neighborhoods of the ith robot. The red curve is the total potential energy received by robot i. When the total potential field is greater than the threshold, the robot enters the obstacle avoidance mode to achieve precise position control, otherwise the robot will enter the cruise mode to prolongs the endurance time.

The communication trigger can help the robot in cruise mode reduce the communication and driving frequency. In cruise mode, the trigger moment depends on the error of the virtual linkage state, which can be denoted as $\boldsymbol{\varepsilon}_l(t) = \widehat{\Delta \boldsymbol{L}}_l(t) - \Delta \boldsymbol{L}_l(t)$, where $\Delta \boldsymbol{L}_l(t) = \boldsymbol{L}_l(t) - \boldsymbol{L}_l^d(t)$ is the error of the virtual link, $\widehat{\Delta \boldsymbol{L}}_l(t) = \Delta \boldsymbol{L}_l\left({}^C t_k^l\right), {}^C t_k^l \in \left[{}^C t_k^l, {}^C t_{k+1}^l\right), k = 0, 1, \ldots..$ is the error at the last trigger moment, t_k^l is the event trigger moment. In order to reduce the communication and driving frequency, the trigger condition is defined as

$$^C f_l\left(t, \boldsymbol{\varepsilon}_l(t)\right) = |\boldsymbol{\varepsilon}_l(t)| - \sigma_l e^{-\gamma t} < 0, \tag{10}$$

where $0 < \sigma_l, 0 < \gamma \leq \max_{2 \leq i \leq N}\left\{\frac{\mathrm{Re}(\lambda_i)}{2}\right\}$. λ_i is the eigenvalues of the Laplacian matrix L. The event trigger condition will be continuously monitored by the communication trigger. For example, when $^C f_l\left(t, \varepsilon_l(t)\right) > 0$, the formation control strategy will receive the state of the neighbor robot and generate a new control vector, otherwise the control vector of the machine will remain unchanged. The trigger moment of the drive trigger can be defined as

$$^C t_{k+1}^l = \inf\left\{t > {}^C t_k^l \mid {}^C f_l\left(t, \boldsymbol{\delta}_l(t)\right) \geq 0\right\}. \tag{11}$$

E. The Design of the Adaptation Formation Law Based on Virtual Linkage

In order to allows the multi-ASR system to pass the restricted region during formation keeping, motivated by the pendulum movement, we designed the adaptive formation control strategy for the multi-ASR system. The movement of the pendulum is mainly affected by gravity and pulling force. In our proposed adaptation law, the control vector generated by the virtual linkage algorithm is assumed to be the gravity and the control vector generated by the obstacle avoidance algorithm is assumed to be the pulling force of the robot. So the multi-ASR system can pass the restricted region like the pendulum by changing the angle of the virtual link.

The proposed strategy is also based on the improved virtual linkage formation strategy, which is shown in Fig. 6. The angle control plane is formed of VL_l and VL_l^d, which can rotate around VL_l^d. The angle control plane simplifies the adaptive pendulum motion of multi-ASR system by transforming its three-dimensional motion into a two-dimensional motion in the plane. Two instances where the adaptive formation control strategy is required to enable multi-ASR system to pass the different region during formation keeping are described in Fig. 8.

In the first case [see Fig. 8(a)], the robot in cruise mode access to continuous measurements and control signals. Only when the state error satisfies the designed triggering condition, the virtual link state and the control signal will be updated, and the control signal is only affected by the desired state. Under the edge-based event-triggered condition (10), for individual robot i, the control protocol is designed as

$$\boldsymbol{F}_i(t) = \boldsymbol{F}_i^g(t) = -\sum_{l=1}^{M} \bar{d}_{il} \widehat{\Delta \boldsymbol{L}}_l(t). \tag{12}$$

In the second case [see Fig. 8(b)], the robot in cruise mode need to achieve the given formation while avoiding collisions with obstacles. When robot enter the cruise mode, the communication and control signals is discrete and its control signals will be affected by obstacles and the desired state. The virtual resultant force of the robot can be defined as

$$\boldsymbol{F}_i(t) = \omega \boldsymbol{F}_i^g(t) + (1-\omega)^l P_{xz} \boldsymbol{F}_i^o(t). \tag{13}$$

$$R = \begin{bmatrix} \cos\left(\sum \alpha_l\right) & \sin\left(\sum \alpha_l\right) & 0 \\ -\cos\left(\sum \beta_l\right)\sin\left(\sum \alpha_l\right) & \cos\left(\sum \beta_l\right)\cos\left(\sum \alpha_l\right) & \sin\left(\sum \beta_l\right) \\ \sin\left(\sum \beta_l\right)\sin\left(\sum \alpha_l\right) & -\sin\left(\sum \beta_l\right)\cos\left(\sum \alpha_l\right) & \cos\left(\sum \beta_l\right) \end{bmatrix}. \tag{6}$$

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

The ω is the weight. The P_{xz} is a projection matrix projected on xz plane. $\boldsymbol{F}_i^g(t)$ is obtained by the improved virtual linkage algorithm, which is regarded as gravity to maintain formation, and $\boldsymbol{F}_i^o(t)$ obtained by equation (9) is regard as the pulling force, which is used for obstacle avoidance. Different from the gravity in cruise mode, the gravity in avoidance mode is a continuous control signal, which is defined as

$$\boldsymbol{F}_i^g(t) = -\sum_{l=1}^{M} \boldsymbol{d}_{il}\, \Delta \boldsymbol{L}_l(t). \tag{14}$$

The pulling force $\boldsymbol{F}_i^o(t)$ can be decomposed into three components, where ${}^x\boldsymbol{F}_i^o(t)$ and ${}^y\boldsymbol{F}_i^o(t)$ are perpendicular to each other and located in the angle control plane, and ${}^z\boldsymbol{F}_i^o(t)$ is perpendicular to the angle control plane. In order to simplify the movement of the robot and make the virtual link converge to the desired state faster, we only consider the influence of ${}^x\boldsymbol{F}_i^o(t)$ and ${}^z\boldsymbol{F}_i^o(t)$ on the virtual link. Then the torque on the virtual link l in the angle control plane can be defined as

$$\boldsymbol{M}_l^\alpha = VL_l \times (1-\omega)^x \boldsymbol{F}_i^o(t). \tag{15}$$

Then the angle variation $\Delta\boldsymbol{\alpha}_l(t)$ of virtual connecting can be defined as

$$\Delta\boldsymbol{\alpha}_l(t) = \int_t^{t+1} \boldsymbol{M}_l^\alpha / \boldsymbol{I}_l dt, \tag{16}$$

where $\boldsymbol{I} = ml^2$ is the moment of inertia of the virtual link l. $\Delta\boldsymbol{\beta}_l(t)$ can be obtained in the same way. Then the desired position of the robot can be obtained by equation (4).

IV. Simulation and Results

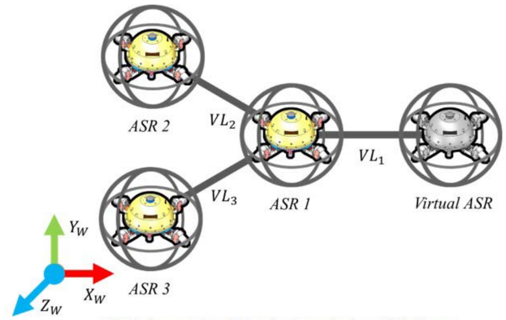
In this section, to verify the effectiveness of the multi-ASR cooperative formation control strategy, a simulation was built in MATLAB. In the experiment, three ASRs and one virtual ASR were used to evaluate the performance of the formation control strategy. These four robots are designed as one virtual linkage containing three virtual links. Fig. 9(a) and Fig. 9(b) are the virtual linkage structure and communication topology diagrams, respectively.

The network topology diagram can effectively describe the communication between ASRs. In the four-ASR topology, a virtual ASR is used to constrain the position and shape of the multi-ASRs system. The four ASRs have the same characteristics. The degree matrix D and the Laplacian matrix L and is given as:

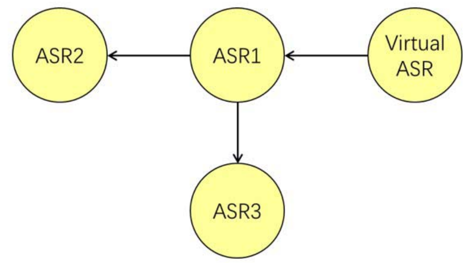
$$D = \begin{bmatrix} 1 & -1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix},$$

$$L = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{17}$$

The desired position of each robot is set to $\boldsymbol{L}_1^d = \left[\begin{bmatrix} 1 & 0 \end{bmatrix} \frac{5\pi}{4} 0\ 1\right]^T$, $\boldsymbol{L}_2^d = \left[\begin{bmatrix} 1 & 0 \end{bmatrix} \frac{3\pi}{2}\ \frac{\pi}{4}\ \sqrt{2}\right]^T$ and $\boldsymbol{L}_3^d = \left[\begin{bmatrix} 2 & 0 \end{bmatrix} \frac{3\pi}{2}\ 0\ 2\right]^T$. The ${}^M\delta_i$ in mode trigger is set as 0.1. The σ_l and γ in communication trigger is defined as 1 and 0.2.

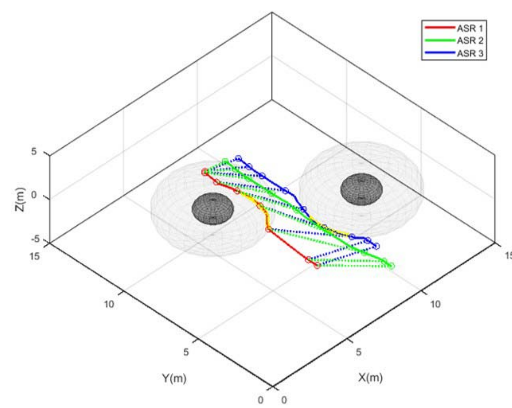


(a) Schematic of the designed virtual linkage

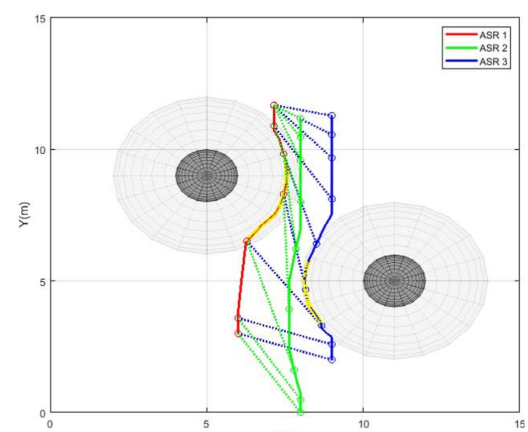


(b) Communication topologies

Fig. 9. Schematic of designed virtual linkage and communication topologies.



(a) Side view: The trajectories of each robot.



(b) Top view: The trajectories of each robot.

Fig. 10. The trajectories of each robot in the simulation.

Two sphere obstacles are considered with centers at $\begin{bmatrix} 5 & 9 & 0 \end{bmatrix}^T$ and $\begin{bmatrix} 11 & 5 & 0 \end{bmatrix}^T$ and the radius of 0.2 m. Simulation results are shown in Fig. 10–Fig. 13.

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

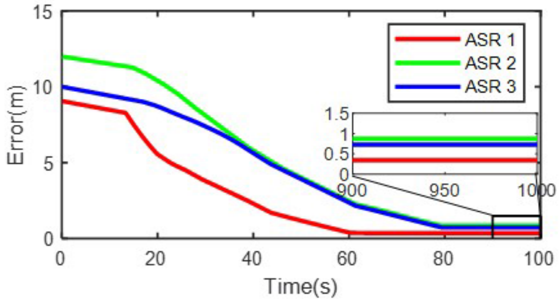


Fig. 11. Formation error in the simulation.

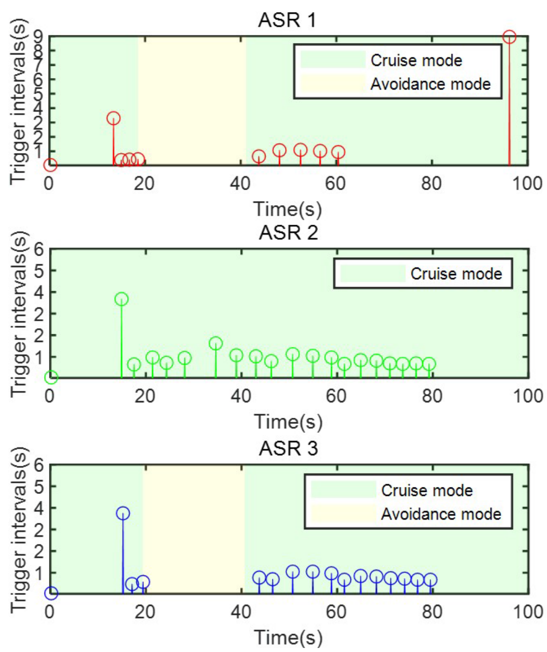
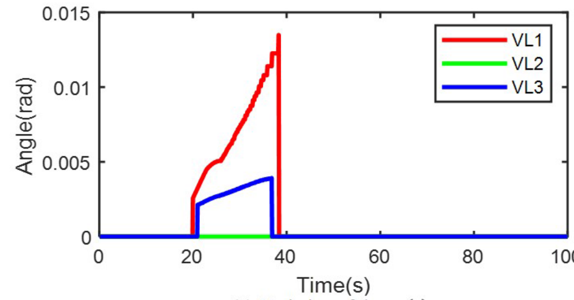


Fig. 12. Trigger intervals of each robot.

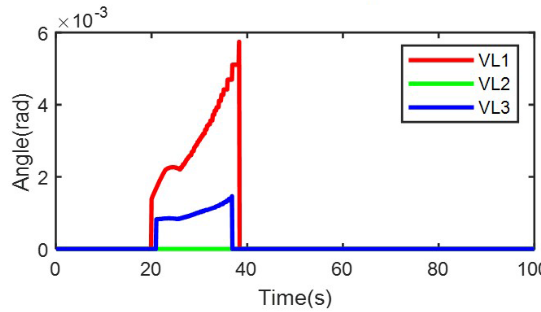
The trajectories are collected and shown in Fig. 10. Fig. 11 shows the formation errors of each ASR. The yellow trajectory line in Fig. 10 represents the ASR is in the avoidance mode. Fig. 10 and Fig. 11 indicate that all robot states can converge to the desired state and adjust their state to avoid obstacles. The trigger intervals are given in Fig. 12. The driving frequency of robot 1, 2 and 3 has been reduced by 98.1%, 98.0% and 97.8% respectively. Fig. 13 describe the variation of the virtual link angle. We know that the robot in avoidance mode can adaptively adjust the state of the virtual link according to environmental information.

V. Experiment on Formation of Multiple ASRs

In this section, the effectiveness and robustness of the proposed formation control strategy are verified through experiments with multi-ASRs platform. The experiment was carried out in the experimental tank with dimensions 4m $\times$ 3.2m $\times$ 1m depicted in Fig. 14. The yaw, pitch and roll angle of the ASR is measured by a high-precision IMU(LORD, 3DM-GX5-45) equipped on the robot with the angle precision of $\pm 0.25°$. The position of ASRs is provided by a camera



(a) Variation of the $\alpha_l(t)$



(b) Variation of the $\beta_l(t)$

Fig. 13. Angular variation of virtual link.

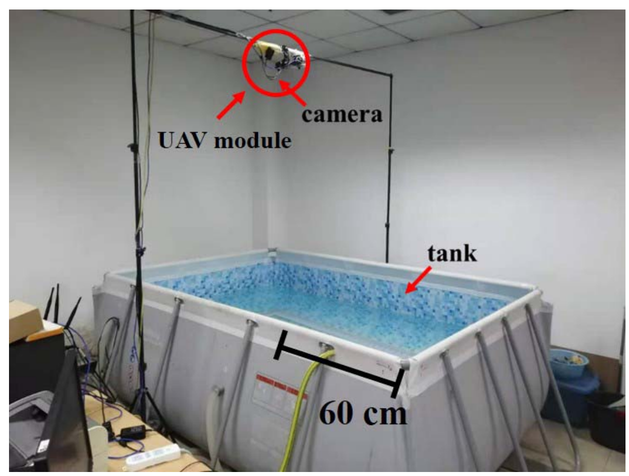


Fig. 14. Experimental setup.

above the pool. All the experiments are run 5 times to ensure repeatability.

In the experiment, two ASRs and one virtual ASR were used to evaluate the performance of the formation control strategy. These three robots are designed as one virtual linkages containing two virtual links. Fig. 15(a) and Fig. 15(b) are the virtual link structure and communication topology diagrams, respectively. The degree matrix D and the Laplacian matrix L and can be written as:

$$D = \begin{bmatrix} 1 & -1 \\ 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad L = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{18}$$

The desired state of the virtual link is set to $\boldsymbol{L}_1^d = \left[\left[\,0.8\ 0\,\right] 3.8\ 0\ 0.8\right]^T$ and $\boldsymbol{L}_2^d = \left[\left[\,0.8\ 0\,\right] 2.48\ 0\ 0.8\right]^T$. Due to the limitation of the experimental environment, the ${}^M\delta_i$ in

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

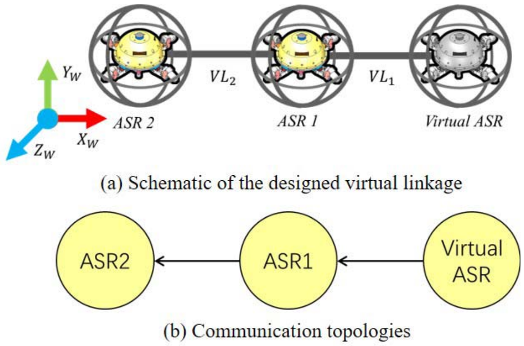


Fig. 15. Schematic of designed virtual linkage and communication topologies.

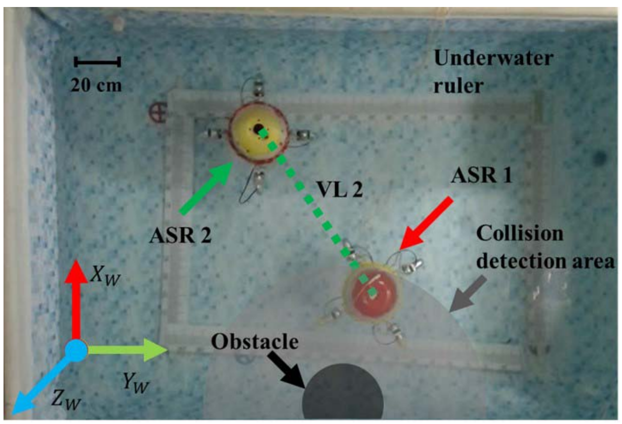


Fig. 16. Formation test of two amphibious spherical robot ASRs.

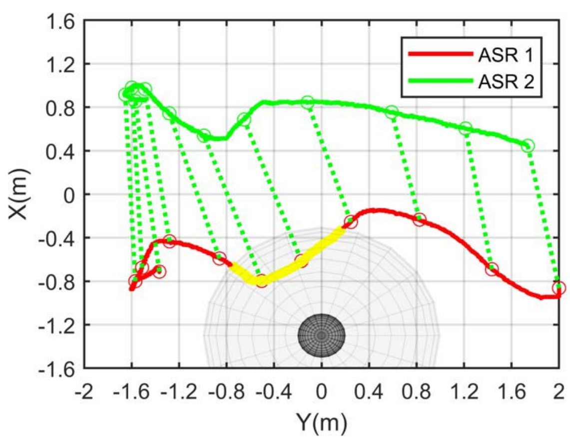


Fig. 17. The trajectories of each robot.

mode trigger is set as 0. The σ_l and γ in communication trigger is defined as 1 and 0.5. A virtual sphere obstacle is considered with centers at $\left[0\ -1.3\ 0\right]^T$ and the radius of 0.2 m. Fig. 16 depicts an experimental scenario in which the two ASRs (labeled robot1 and robot2) are preparing to converge to the desired state while avoiding a virtual obstacle. As shown in Fig. 16, ASR1 was in avoidance mode and was avoiding the virtual spherical obstacle. ASR2 was in cruise mode and followed ASR1 under the control of the virtual link VL_2. The results are shown in Fig. 17–Fig. 20.

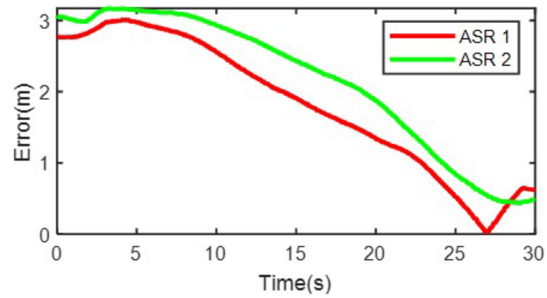


Fig. 18. Formation error in the experiment.

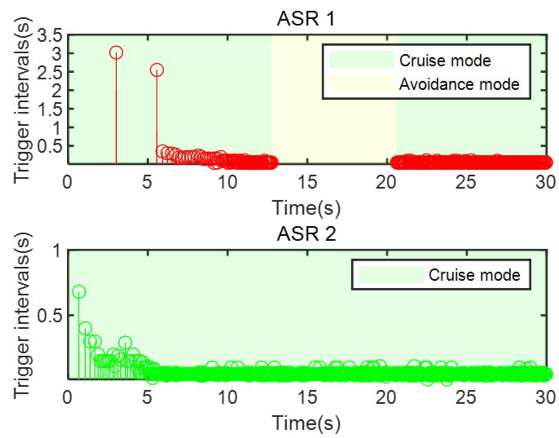


Fig. 19. Trigger intervals of each robot.

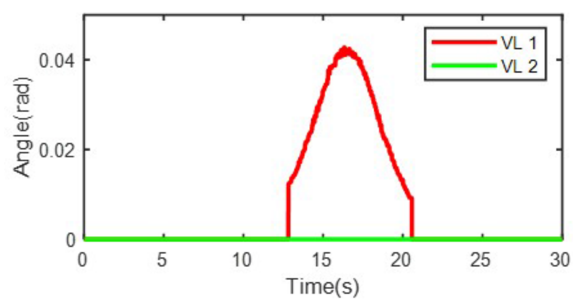


Fig. 20. The variation of angle $\alpha_l(t)$ on virtual link.

Fig. 17 shows the trajectories of all robots, from which we can see that all virtual links can converge to the desired state by locally adjusting the self-state to avoid collisions with obstacles. The yellow trajectory line in Fig. 17 represents the ASR is in avoidance mode and it indicates that the ASR can switch the motion mode according to the environment information. Fig. 18 shows the formation error of all robots. The trajectory error at the last moment in Fig. 18 is due to the interference of the tank wall to the ASR. The trigger intervals of all robots are shown in Fig. 19. Compared with the traditional continuous formation control algorithm, the driving frequency of robot 1 and 2 has been reduced by 87.2%, and 84.1% respectively. The result also indicates that the robot can choose different motion modes based on environmental information. When the robot is in avoidance mode, the event interval is 0; when the robot is in cruise mode, the event

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

interval is positive. The angle variation is shown in Fig. 20. It can be observed that the formation of multi-robot formations can change when the robots pass obstacles. All results prove the feasibility of the proposed control strategy.

VI. Conclusion

In this paper, we have presented a formation control strategy that enables multiple ASRs to form a desired virtual linkage and avoid collisions with limited communication and driving frequency. First, the virtual linkage formation strategy is extended to propose a three-dimensional formation strategy based on the virtual linkage for multi-ASR system. Though the design of mode triggers and communication triggers based on event trigger condition of virtual linkage state, the proposed formation control strategy allows the multi-ASR system to maintain formation and avoid obstacles with limited communication and driving frequencies. In addition, inspired by the pendulum movement, an adaptation law has been developed to adjust the given angle of the virtual linkage in the case that the restricted path is impassable for the desired formation. The simulation and the experiment integrated into the multi-ASR system verified the effectiveness of the proposed formation control strategy. we believe that the proposed formation control strategy will help robots improve the efficiency of exploration in complex amphibious environments.

References

[1] H. Xing, S. Guo, L. Shi, X. Hou, Y. Liu, and H. Liu, "Design, modeling and experimental evaluation of a legged, multi-vectored water-jet composite driving mechanism for an amphibious spherical robot," *Microsyst. Technol.*, vol. 26, no. 2, pp. 475–487, Feb. 2020.

[2] R. An, S. Guo, Y. Yu, C. Li, and T. Awa, "Task planning and collaboration of jellyfish-inspired multiple spherical underwater robots," *J. Bionic Eng.*, vol. 19, no. 3, pp. 643–656, May 2022.

[3] L. Ma, Z. Wang, Q.-L. Han, and Y. Liu, "Consensus control of stochastic multi-agent systems: A survey," *Sci. China Inf. Sci.*, vol. 60, no. 12, pp. 1–15, Dec. 2017.

[4] F. Liao, R. Teo, J. L. Wang, X. Dong, F. Lin, and K. Peng, "Distributed formation and reconfiguration control of VTOL UAVs," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 1, pp. 270–277, Jan. 2016.

[5] F. Shkurti *et al.*, "Underwater multi-robot convoying using visual tracking by detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 4189–4196.

[6] W. Wang *et al.*, "Roboat II: A novel autonomous surface vessel for urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 1740–1747.

[7] H. Rezaee and F. Abdollahi, "A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots," *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 347–354, Jan. 2013.

[8] G. Wen, C. L. P. Chen, and Y.-J. Liu, "Formation control with obstacle avoidance for a class of stochastic multiagent systems," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5847–5855, Jul. 2018.

[9] L. Dong, Y. Chen, and X. Qu, "Formation control strategy for nonholonomic intelligent vehicles based on virtual structure and consensus approach," *Proc. Eng.*, vol. 137, pp. 415–424, Jan. 2016.

[10] X. Li and D. Zhu, "An adaptive SOM neural network method for distributed formation control of a group of AUVs," *IEEE Trans. Ind. Electron.*, vol. 65, no. 10, pp. 8260–8270, Oct. 2018.

[11] Y. Liu, J. Gao, X. Shi, and C. Jiang, "Decentralization of virtual linkage in formation control of multi-agents via consensus strategies," *Appl. Sci.*, vol. 8, no. 11, p. 2020, Oct. 2018.

[12] X. Liu, S. S. Ge, C.-H. Goh, and Y. Li, "Event-triggered coordination for formation tracking control in constrained space with limited communication," *IEEE Trans. Cybern.*, vol. 49, no. 3, pp. 1000–1011, Mar. 2019.

[13] F. Berlinger, M. Gauci, and R. Nagpal, "Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm," *Sci. Robot.*, vol. 6, no. 50, Jan. 2021, Art. no. eabd8668.

[14] L. Ding, Q.-L. Han, X. Ge, and X.-M. Zhang, "An overview of recent advances in event-triggered consensus of multiagent systems," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1110–1123, Apr. 2018.

[15] C. Nowzari, E. Garcia, and J. Cortés, "Event-triggered communication and control of networked systems for multi-agent consensus," *Automatica*, vol. 105, pp. 1–27, Jul. 2019.

[16] A. Vega-Alonzo, J. Guerrero-Castellanos, S. Durand, N. Marchand, S. Maya-Rueda, and G. Mino-Aguilar, "Event-based control strategy for consensus of a group of VTOL-UAVs," in *Proc. 2nd Int. Conf. Event-based Control, Commun., Signal Process. (EBCCSP)*, 2016, pp. 1–8.

[17] W. Song, J. Wang, S. Zhao, and J. Shan, "Event-triggered cooperative unscented Kalman filtering and its application in multi-UAV systems," *Automatica*, vol. 105, pp. 264–273, Jul. 2019.

[18] D. Jang, C. Y. Son, J. Yoo, H. J. Kim, and K. H. Johansson, "Efficient networked UAV control using event-triggered predictive control," *IFAC-PapersOnLine*, vol. 52, no. 15, pp. 412–417, 2019.

[19] X. Ge and Q.-L. Han, "Distributed formation control of networked multi-agent systems using a dynamic event-triggered communication mechanism," *IEEE Trans. Ind. Electron.*, vol. 64, no. 10, pp. 8118–8127, Oct. 2017.

[20] L. Zhang, J. Sun, and Q. Yang, "Distributed model-based event-triggered leader-follower consensus control for linear continuous-time multiagent systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 99, pp. 1–9, Jan. 2020.

[21] F. Xiao, T. Chen, and H. Gao, "Synchronous hybrid event- and time-driven consensus in multiagent networks with time delays," *IEEE Trans. Cybern.*, vol. 46, no. 5, pp. 1165–1174, May 2016.

[22] P. Ming, J. Liu, S. Tan, S. Li, L. Shang, and X. Yu, "Consensus stabilization in stochastic multi-agent systems with Markovian switching topology, noises and delay," *Neurocomputing*, vol. 200, pp. 1–10, Aug. 2016.

[23] L. Ding and W. X. Zheng, "Consensus tracking in heterogeneous nonlinear multi-agent networks with asynchronous sampled-data communication," *Syst. Control Lett.*, vol. 96, pp. 151–157, Oct. 2016.

[24] J. Zhou, Y. Guo, G. Li, and J. Zhang, "Event-triggered control for nonlinear uncertain second-order multi-agent formation with collision avoidance," *IEEE Access*, vol. 7, pp. 104489–104499, 2019.

[25] M. Longinos-Garrido, N. A. Tolentino-Medrano, J. F. Guerrero-Castellanos, J. Castañeda-Camacho, and R. Álvarez González, "Event-triggered coordination of omnidirectional robots over gazebo," in *Proc. 6th Int. Conf. Event-Based Control, Commun., Signal Process. (EBCCSP)*, 2020, pp. 1–8.

[26] P. Singh *et al.*, "Multivariable event-triggered generalized super-twisting controller for safe navigation of nonholonomic mobile robot," *IEEE Syst. J.*, vol. 15, no. 1, pp. 454–465, Mar. 2020.

[27] F. Huang, P. Wu, and X. Li, "Distributed flocking control of quad-rotor UAVs with obstacle avoidance under the parallel-triggered scheme," *Int. J. Control, Autom. Syst.*, vol. 19, no. 3, pp. 1375–1383, Mar. 2021.

[28] Y. He, L. Zhu, G. Sun, J. Qiao, and S. Guo, "Underwater motion characteristics evaluation of multi amphibious spherical robots," *Microsyst. Technol.*, vol. 25, no. 2, pp. 499–508, Feb. 2019.

[29] X. Hou *et al.*, "Hydrodynamic analysis-based modeling and experimental verification of a new water-jet thruster for an amphibious spherical robot," *Sensors*, vol. 19, no. 2, p. 259, Jan. 2019.

[30] H. Xing *et al.*, "Hybrid locomotion evaluation for a novel amphibious spherical robot," *Appl. Sci.*, vol. 8, no. 2, p. 156, Jan. 2018.

[31] H. Xing *et al.*, "Robust RGB-D camera and IMU fusion-based cooperative and relative close-range localization for multiple turtle-inspired amphibious spherical robots," *J. Bionic Eng.*, vol. 16, no. 3, pp. 442–454, May 2019.

[32] B. Wei, F. Xiao, and M.-Z. Dai, "Edge event-triggered control for multi-agent systems under directed communication topologies," *Int. J. Control*, vol. 91, no. 4, pp. 887–896, 2018.

[33] X. Xie, Q. Zhou, D. Yue, and H. Li, "Relaxed control design of discrete-time Takagi–Sugeno fuzzy systems: An event-triggered real-time scheduling approach," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 12, pp. 2251–2262, Dec. 2018.

[34] H. Lu, X. Xie, and C. Peng, "Relaxed conditions of observer design of discrete-time Takagi–Sugeno fuzzy systems via a new multi-instant gain-scheduling scheme," *IEEE Trans. Fuzzy Syst.*, early access, Jul. 7, 2021, doi: 10.1109/TFUZZ.2021.3095387.

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.

He Yin (Student Member, IEEE) received the B.Eng. and M.Eng. degrees in biomedical engineering from the Beijing Institute of Technology, Beijing, China, in 2017 and 2019, respectively, where he is currently pursuing the Ph.D. degree.

His current research interests include amphibious robot and multi-robot cooperation.

Shuxiang Guo (Fellow, IEEE) received the Ph.D. degree in mechano-informatics and systems from Nagoya University, Nagoya, Japan, in 1995.

He is currently the Chair Professor with the Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, Ministry of Industry and Information Technology, Beijing Institute of Technology, Beijing, China. He has published about 710 refereed journals and conference papers. His research interests include micro-catheter systems, medical robot systems for minimal invasive surgery, biomimetic underwater robots, and smart material (SMA and IPMC) based on actuators.

Prof. Guo is the Editor-in-Chief of the *International Journal of Mechatronics and Automation*.

Meng Liu (Student Member, IEEE) received the B.S. degree in control theory and applications from the Beijing Institute of Technology, Beijing, China, in 2015, where he is currently pursuing the M.S. degree in biomedical engineering with the Key Laboratory of Convergence Medical Engineering System and Healthcare Technology.

His current research interests include trajectory tracking and avoiding obstacle on amphibious spherical robot.

Authorized licensed use limited to: BEIJING INSTITUTE OF TECHNOLOGY. Downloaded on July 04,2022 at 01:25:54 UTC from IEEE Xplore. Restrictions apply.