Journal of Marine Science and Application (2019) 18:343–352
https://doi.org/10.1007/s11804-019-00097-3

RESEARCH ARTICLE

Development of an Inexpensive Decentralized Autonomous Aquatic Craft Swarm System for Ocean Exploration

Runlong Miao[1] · Shuo Pang[1] · Dapeng Jiang[1]

Received: 23 March 2018 / Accepted: 16 January 2019 / Published online: 16 August 2019

Abstract
Swarm robotics in maritime engineering is a promising approach characterized by large numbers of relatively small and inexpensive autonomous aquatic crafts (AACs) to monitor marine environments. Compared with a single, large aquatic manned or unmanned surface vehicle, a highly distributed aquatic swarm system with several AACs features advantages in numerous real-world maritime missions, and its natural potential is qualified for new classes of tasks that uniformly feature low cost and high efficiency through time. This article develops an inexpensive AAC based on an embedded-system companion computer and open-source autopilot, providing a verification platform for education and research on swarm algorithm on water surfaces. A topology communication network, including an inner communication network to exchange information among AACs and an external communication network for monitoring the state of the AAC Swarm System (AACSS), was designed based on the topology built into the Xbee units for the AACSS. In the emergence control network, the transmitter and receiver were coupled to distribute or recover the AAC. The swarm motion behaviors in AAC were resolved into the capabilities of go-to-waypoint and path following, which can be accomplished by two uncoupled controllers: speed controller and heading controller. The good performance of velocity and heading controllers in go-to-waypoint was proven in a series of simulations. Path following was achieved by tracking a set of ordered waypoints in the go-to-waypoint. Finally, a sea trial conducted at the China National Deep Sea Center successfully demonstrated the motion capability of the AAC. The sea trial results showed that the AAC is suited to carry out environmental monitoring tasks by efficiently covering the desired path, allowing for redundancy in the data collection process and tolerating the individual AACs' path-following offset caused by winds and waves.

Keywords Marine environment monitoring · Swarm robotics · Autonomous aquatic craft · Unmanned surface vehicles · Autonomous aquatic craft swarm system · Decentralized control

Article Highlights
- Developing an inexpensive AAC based on an embedded-system companion computer and open-source autopilot.
- Providing a verification platform for education and research on swarm algorithm on water surfaces.
- Integrating an inner communication network to exchange information among AACs with an external communication network for monitoring the state of the AACSS.
- Allowing for redundancy in the data collection process and tolerating the individual AACs' path-following offset caused by winds and waves.

✉ Shuo Pang
sspp27@hotmail.com

[1] National Key Laboratory of Autonomous Underwater Vehicle, College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China

1 Introduction

Maritime tasks for collecting large amounts of spatially and temporally dispersed data are usually expensive to carry out due to the use of manned vehicles (Wright and Baldauf 2016) with large operational crews or large unmanned expensive vehicles (Dunbabin and Marques 2012) requiring various complex equipment. In Leonard et al. (2010), six relatively simple and inexpensive autonomous gliders carrying environmental sensors were autonomously coordinated to sample the ocean for 24 days. Valada et al. (2014), on the other hand, developed a low-cost fleet of vessels that could monitor water quality in several large areas. These studies saved considerable money in marine tasks by replacing one large vehicle with a group of small ones. However, these vehicles are based on centralized path-planning solutions and require additional command from a base station to guide a mission. Furthermore, centralized control (Crespi et al. 2008) is subject to computational and/or communication constraints dealing with mass data and lacks scalability and fault tolerance. Decentralized control, on the other hand, solves the problem of a single point of failure and processes mass data in a decentralized system (Brambilla et al. 2013), and once coordination is achieved among autonomous aquatic crafts (AACs) through a swarm network, decentralized control is scaled well, similar to the works of Christensen et al. (2009) and Wolf et al. (2017).

If people could find a strategy that can undertake the same tasks with low cost, high efficiency, and less manual intervention, then more individuals will achieve the opportunities to explore the ocean. The AAC Swarm System (AACSS), which comprises a monitor and more than two AACs with decentralized control, is a novel swarm robotics approach to meet this demand. Xu et al. (2014) applied a large number of relatively simple wireless sensor nodes to collect data across multiple sites in parallel. Duarte et al. (2016) synthesized eight aquatic swarm robots to collect temperature data uniformly through time in a semi-enclosed area in the margins of the Tagus River. However, these nodes possessed no motion capability, and the swarm robots are unstable when sailing in the sea with waves and winds.

This article presents the design and development of an AAC with a swarm communication network and decomposes all the swarm motion behaviors, such as aggregation (Soysal and Bahç 2007; Yu et al. 2017), flocking (Zhu et al. 2017), foraging (Castello et al. 2016), and formation (Dong et al. 2016) on the water surface, into two basic capabilities of go-to-waypoint and path following for a single AAC. For the economically feasible deployment of a large number of units for the AACSS, the cost of each unit must be kept low, implying that a single AAC must be kept simple, easy to be manufactured, and perform well in real world. The major design criteria of this AAC are as follows: (1) it must be relatively simple and low cost, reducing total cost and difficulty to manufacture the AACSS; (2) each unit must be relatively small but large enough to hold all the payload and withstand winds and waves, allowing for a portable deployment and excellent maneuverability; (3) each unit must be capable of obtaining its location and attitude information, autonomous decision-making, and collaborating with other units; (4) the AACSS provides the users an intuitive and easy-to-use command and control interface.

In the following sections, we provide an overview of the design and manufacturing of the AAC (Section 2), a description of the AAC's speed and heading controller (Section 3), the AAC performance tests conducted in simulation and sea trial with discussion (Section 4), and concluding remarks (Section 5).

2 Autonomous Aquatic Craft Design and Software

Every component was selected with consideration for both functionality and cost. Especially, the fishing boat attached with two thrusters, the enclosure, and the electronics are relatively cheap from online stores, resulting in a total cost a bit over 500 US dollars (details shown in Table 1) and a short design-to-product cycle. In total, three students manufactured 15 AACs in less than 1-month time frame.

2.1 Boat Components

Considering the design criteria above, the main body of a single AAC originated from an off-the-shelf fishing bait boat (see:https://item.taobao.com/item.htm?spm=a230r.1.14.254.7924554crALC4J&id=583111634542&ns=1&abbucket=19#detail), and all hardware and software were based on open-source drone systems. Although the boat shown in Fig. 1 is relatively small (L 56 cm × W 27 cm × H 26 cm) and light (2.5 kg), it features a (L 21 cm × W 15 cm × H 10 cm) cuboid space to hold all the electronic components. Compared with the same size of a mono-hull boat, this catamaran also exhibits certain advantages, such as good seakeeping, turning capability, and sufficient layout space for more sensors, which would benefit the subsequent research. Most of the electronic components were housed in a compact electronic enclosure isolated from harsh environmental elements, such as heat and salt water. The upper enclosure was produced using a 3D printer and installed on the top of the boat using epoxy resin.

2.2 Electronic Components

The electronic component architecture was mainly composed of the electronic system on board and monitor on the shore and emergence control (see Figs. 2 and 3). A transmitter and a

Table 1 Itemized swarm unit budget

Component	Quantity	Unit price ($)	Cost ($)
Main hull	1	196.1	196.1
Upper enclosure	1	3.1	3.1
Raspberry PI 3B (RPI)	1	32.2	32.2
Pixhawk 2.4.6	1	53.6	53.6
HobbyWing 1060 ESC	2	13.8	27.6
M8N GPS	1	32.2	32.2
Xbee Pro 900HP	2	44.4	88.8
Xbee antenna	2	3.1	6.2
IP67 connector	4	2.7	10.8
Battery	2	33.6	67.2
Other	1	20	20
Total Cost	537.8		

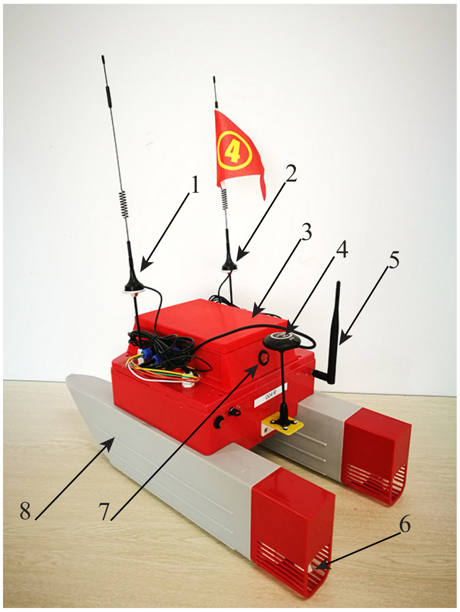


Fig. 1 AAC (the first antenna on the starboard is connected with the external network, the second antenna on the port is connected with the inner network, and the third antenna is used for emergence control). 1, inner network antenna; 2, external network antenna; 3, electronic enclosure; 4, GPS; 5, emergence control antenna; 6, thruster; 7, catamaran hull; 8, power switch

laptop with a Xbee module were set on the base station to manually distribute or recover the AAC in the case of emergency and monitor the state of AAC, separately. The electronic system placed in the AAC's main body performs two roles: to collect the urgent order from the transmitter via a receiver, and to execute the swarm algorithm and then output the motion command to the two thrusters. A relay serves as a switch between these options, which is under the command of the emergence control. The bottom layer possesses two thrusters which provide the propulsion system coupled with two electronic speed controllers (ESCs).

Each AAC should be capable of autonomous decision-making (Parker and Zhang 2011), which is a necessary condition for the collective decision-making of AACSS. This goal was achieved through the inclusion of an onboard processing computer, an autopilot unit, sensors, and communication devices. The swarm algorithm runs on a single-board computer, which can perform computationally intensive or time-sensitive tasks. A Raspberry PI 3B (RPI) was selected as the onboard computer (Danymol et al. 2014), which is connected to the autopilot unit Pixhawk through a USB port. The Pixhawk (Meier et al. 2012) is an independent, open-hardware project (see https://pixhawk.org/) that aims at providing high-end autopilot at low costs and high availability. This board contains and executes the code required for the autopilot to function correctly. The board also obtains information regarding the position and attitude data from sensors (such as GPS, compass, and gyroscope), performs the motion control loops, and outputs the corresponding pulse–width modulation (PWM) signal to the thrusters. The GPS module and HMC5883L compass were placed on a holder outside the main enclosure. Thereby, they can be isolated from the electromagnetic interference from the thrusters or other components in the electronic enclosure. In the next stage of the project, more sensors, such as temperature or turbidity sensor, were added to each AAC.

2.3 Propulsion and Power System

The propulsion system of a single AAC features a differential drive configuration composed of two thrusters (two ESCs, two motors, and two 3-blade 36-mm plastic propellers). The brush direct-current motors include 12-V input voltage and a maximum 8500r/min rotational speed output. They are driven by two HobbyWing 1060 brushed ESCs, which are controlled by the receiver in the emergent control mode or the Pixhawk autopilot unit in swarm algorithm mode with PWM signals (1500 Hz to 2000 Hz). The two thrusters can approximately linearly generate a propulsive force of up to 12 N in the hydrostatic tests. Given the low cost and relatively high power density, two ACE 4000-mAh 25C 11.1-V lithium–polymer batteries were selected as the power source for the AAC. The endurance of AAC was between 40 min and 1 h depending on the electronic component usage with two batteries.

2.4 Swarm Topology Communication Network

The swarm network can coordinate AACs to reach the target points in higher intelligence and enable data collection across multiple sites in parallel(Şahin 2004). According to the survey of recent advances in environmental analysis (Ballesteros Gómez and Rubio 2009), wireless networks of sensors distributed throughout the environment have recently emerged as a promising technology for marine environmental monitoring and information interchange between surface vehicles. This approach allows for real-time measurement and/or monitoring in locations that are potentially challenging to access.

The simple means of wireless communication (such as equipment based on sound or color) used in swarm network are relatively straightforward to implement in real hardware (Floreano et al. 2007), but they are also limited to exchange information among AACs and prone to be disturbed in a complex ocean environment. To communicate with other AACs and share information with the monitoring station on the shore, each AAC was equipped with two PRO 900PH Xbee wireless communication modules connected to the RPI through USB ports. One Xbee (denoted as Xbee-I in Fig. 2) module was defined as the inner network (see Fig. 4a, b) based

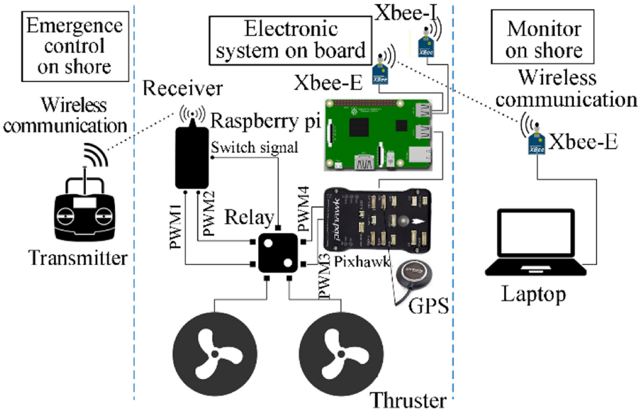


Fig. 2 Electronic components architecure

on ad hoc wireless technology to exchange state messages, including heading, battery status, GPS position, and speed among all AACs for individual AAC autonomous decision-making, whereas the other one (denoted as Xbee-E in Fig. 2) was defined as the external network (see Fig. 4c) to monitor the AACSS status for the base station. Both networks can work individually or standby for each other to improve the reliability of information transmission. The Xbee modules were connected to two monopole antennas fixed on the starboard and port of the AAC, providing an effective communication range from 0 to 800 m. In Fig. 4a, the distances between any two are shorter than 800 m. Thus, all AACs can exchange information with each other. However, the number 2 AAC in Fig. 4b was out of the radio communication coverage of number 3 AAC. Therefor, they communicated with each other through the relay of number 1, 4, and 5 AACs. As shown in Fig. 4c, the laptop serves the central node for collecting information from all the AACs via the external point-to-point network composed of Xbee units labeled as Xbee-E. Every AAC was equipped with a transmitter on shore and a receiver on board to manually distribute or recover the AAC in the case of emergence as shown in Fig. 4d.

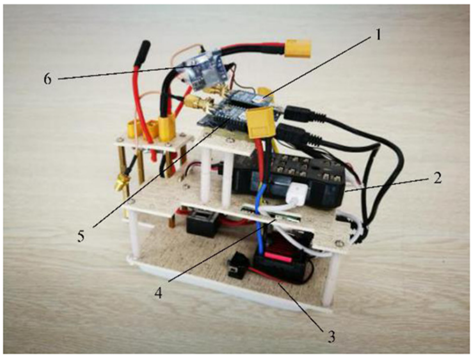


Fig. 3 Electronic components on board. 1, Xbee of inner network; 2, Pixhawk; 3, ESC; 4, RPI; 5, Xbee of external network; 6, voltage and current sensors

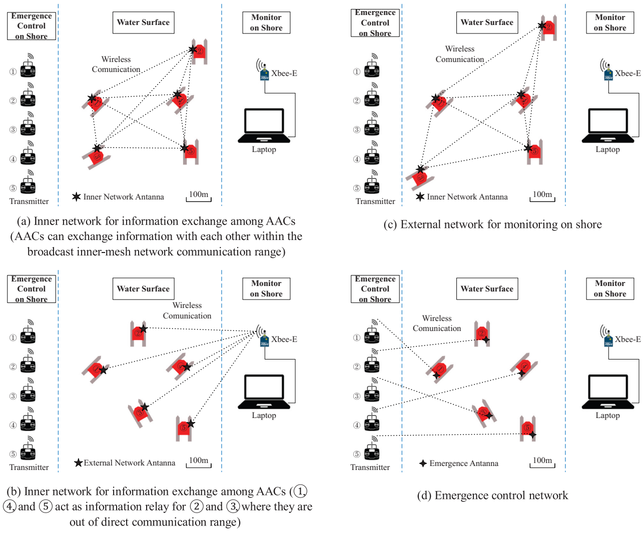
2.5 Autonomous Aquatic Craft Software

The swarm algorithm based on Python with DroneKit–Python API (see http://python.dronekit.io/) running on RPI allows communication through the inner network between the AACs and calculates the desired waypoints. To interact with Pixhawk, inner network, and external network, several open-source software packages (such as mavproxy and pyserial) were installed on RPI. The Pixhawk autopilot determines the desired waypoint and attitude information collection from the sensors attached to it.

A user-friendly human–machine interface Mission Planner (see http://ArduPilot.org/planner) software was developed on a laptop on shore; it allows the operators to supervise the state of AACSS via an external network. Through the same network and with the help of putty software (see https://www.putty.org/), the AAC’s onboard motion control parameters could be further updated or modified. This condition could provide rapid iteration and optimization for swarm algorithm in real-world environments.

3 Controller Design

The capability and efficiency of AACSS depend on the swarm network communication (Li et al. 2008) and the decentralized speed controller and heading controller on individual AAC; these variables are derived from the swarm motion behaviors (Fig. 5). This section aims to develop a simple and universal controller to drive the AAC. The controller structure of every AAC consisted of a navigator, a speed controller, and a heading controller. A simple saturation function and proportional–integral (PI)



(a) Inner network for information exchange among AACs (AACs can exchange information with each other within the broadcast inner-mesh network communication range)

(c) External network for monitoring on shore

(b) Inner network for information exchange among AACs (①④and ⑤act as information relay for ②and ③where they are out of direct communication range)

(d) Emergence control network

Fig. 4 AACSS topology communication network

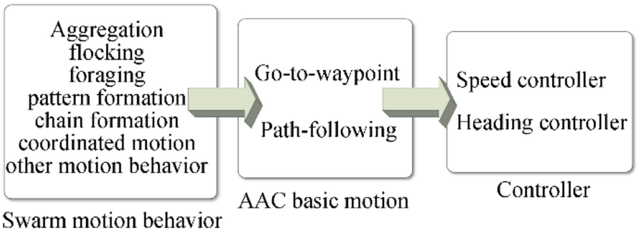


Fig. 5 Simplified swarm behaviors leading to speed controller and heading controller

controller were applied to design the control system of forward speed and heading for the AAC, respectively.

3.1 Autonomous Aquatic Craft Dynamic Model

The three-degree freedom planar model of the AAC (Fig. 6) was adopted, and no coupling existed between the surge and yaw. -F_{L} and -F_{R} are considered as propeller force inputs which push the model to surge, sway, and yaw. The geometrical relationship between the inertial reference frame and the body-fixed frame is denoted in terms of velocities as follows:

$$\begin{cases} \dot{x} = v_x \cos\varphi - v_y \sin\varphi \\ \dot{y} = v_x \sin\varphi + v_y \cos\varphi \\ \dot{\varphi} = r \end{cases} \tag{1}$$

where x, y, and φ denote the position of the center of mass and the orientation angle of the AAC in the inertial reference frame. v_x, v_y, and r are the surge, sway, and yaw velocity, respectively.

In the body-fixed frame, the non-linear equations of motion for a simplified model of the AAC dynamics, where motion in heave, roll, and pitch is neglected, are given by the following:

$$\begin{cases} m_{11}\dot{v}_x - m_{22}\dot{v}_y r + b_1 v_x^{\alpha_1} = F \\ m_{33}\dot{r} + m_d v_x v_y + b_2 sgn(r)|r|^{\alpha_2} = T \end{cases} \tag{2}$$

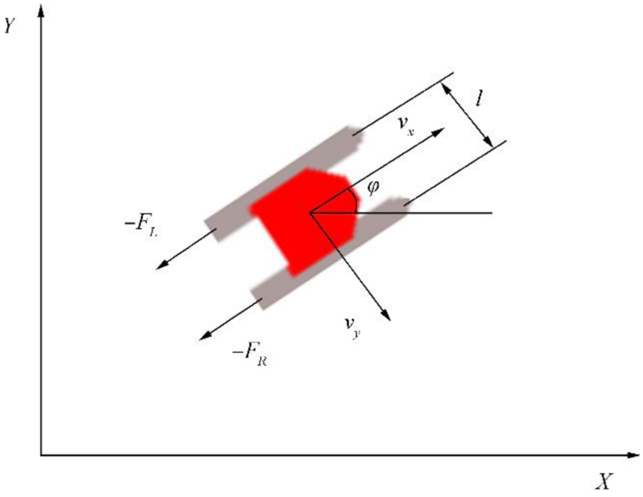


Fig. 6 Dynamic model of the AAC

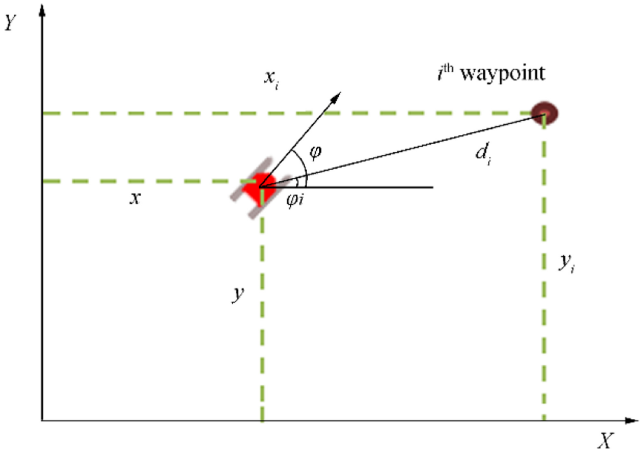


Fig. 7 Geometry definition for waypoint following

where m_{ii} ($i = 1, 2, 3$) are the mass and inertial parameters, $m_d = m_{22} - m_{11} > 0$; F is the surge control force; T is the yaw control moment; and b_1, b_2, α_1, and α_2 refer to the resistance coefficients of surge and yaw motion. In this work, only forward motion was considered, $v_x > 0$, $v_y = 0$, as the reverse motion dynamics differ, and lateral motion is negligible compared with the forward motion. A more detailed discussion of the general spatial equations for the AAC can be found in the work of Fossen (1994). Equation (2) can be written as follows:

$$\dot{v}_x = \left(F - b_1 v_x^{\alpha_1}\right)/m_{11} \tag{3}$$

$$\dot{r} = \left(T - b_2 sgn(r)|r|^{\alpha_2}\right)/m_{33} \tag{4}$$

The surge control force F and the yaw control moment T are given in terms of the two propeller forces as follows:

$$\begin{cases} F = F_{\mathrm{R}} + F_{\mathrm{L}} \\ T = (F_{\mathrm{R}} - F_{\mathrm{L}}) \cdot l/2 \end{cases} \tag{5}$$

where l is the lateral distance between the thrusters.

3.2 Navigation Model

The navigator considers the AAC's current position, which is given by the GPS module, and the desired waypoint to

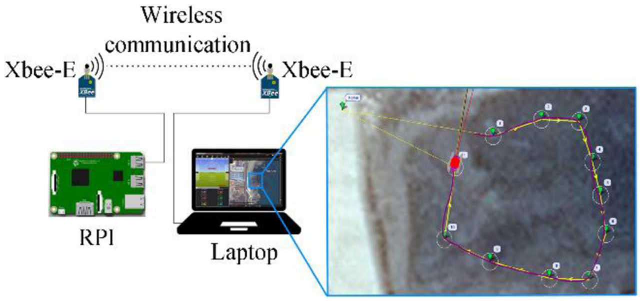


Fig. 8 Simulation system

Table 2 Parameters set in simulation

Parameters	Value	Parameters	Value
α	0.01	b_2	0.002
α_1	1	m_{11} (kg)	4
α_2	1	m_{33} (kg m^2)	0.028
b_1	0.001	U (m/s)	1.5

calculate the distance based on Eq. (6). This approximation is valid as the distance is relatively small for the AAC as shown in Fig. 7.

$$d_i = \sqrt{(x-x_i)^2 + (y-y_i)^2} \tag{6}$$

From the problem described in Fig. 7, geometry yields the following:

$$\varphi_i = arctan\frac{y_i-y}{x_i-x} \tag{7}$$

The target heading φ_i was computed by calculating the line between positions of the AAC and the *i*th waypoint relative to the *x* axis in inertial frame, which is positive in the clockwise direction from *x* axis.

3.3 Speed Controller

The controller was used to change the surge speed or throttle value according to the distance to the *i*th waypoint.

$$v_x = U\left(1-e^{-\alpha(d_i-d_0)}\right) = U\left(1-e^{-\alpha\left[\sqrt{(x-x_i)^2+(y-y_i)^2}-R_0\right]}\right) \tag{8}$$

where *U* denotes the maximum resultant speed of the AAC, α represents the adjustable parameter, and R_0 corresponds to the radius of the waypoint circle.

The AAC was considered to have reached the *i*th waypoint once the distance d_i is less or equal to a certain threshold, and when its speed has decreased rapidly. Care must be taken when defining the threshold with a suitable α as motion conditions (including wind speed) might impede the AAC from crossing it.

After including Eq. (3) and Eq. (5) to Eq. (8), the speed control law is given as follows:

$$F_{\mathrm{R}} + F_{\mathrm{L}} = Um_{11}\left(1-e^{-\alpha\left[\sqrt{(x-x_i)^2+(y-y_i)^2}-R_0\right]}\right)' + b_1 v_x^{\alpha_1} \tag{9}$$

where $b_1 v_x^{\alpha_1}$ refers to the longitudinal resistance, which can be obtained from the static water resistance test or empirical formula.

3.4 PI Heading Controller

Based on Eq. (7), a better practice is to use the function "atan2" as it would also give the quadrant at which the *i*th waypoint is located from the AAC axis:

$$e(t) = \varphi(t)-\varphi_i = \mathrm{atan2}\left(\frac{y(t)-y_i}{x(t)-x_i}\right) \tag{10}$$

where $\varphi(t)$ denotes the AAC heading, which is positive in the clockwise direction from the *x* axis.

The solution is given in radians from π to -π. Further manipulation might be necessary depending on the axis being used.

The PI control algorithm continuously calculates an error value *e*(*t*) and applies a correction based on the proportional and integral terms. The controller attempts to minimize the error over time by adjustment of a control variable *r*(*t*), and the new value is determined by a weighted sum:

$$r(t) = K_{\mathrm{P}} e(k\Delta t) + K_{\mathrm{I}} \sum_{k=0}^{N} e(k\Delta t)\Delta t \tag{11}$$

where K_{P} and K_{I} represent all the non-negatives and denote the coefficients for the proportional and integral terms, respectively (occasionally denoted as *P* and *I*). As the PI controller relies only on the measured process variable and not on knowledge of the underlying process, it is broadly applicable.

By combing Eq. (4), Eq. (5), and Eq. (11), the heading control law can be written as follows:

$$F_{\mathrm{R}}-F_{\mathrm{L}} = \frac{2}{l}\left[m_{33} r'(t) + b_2 sgn(r)|r|^{\alpha_2}\right] \tag{12}$$

where $b_2 sgn(r)|r|^{\alpha_2}$ is the rotation resistance, which can be obtained from the static water resistance test or empirical formula.

4 Tests and Results

The researchers set 11 waypoints (a circular region with a radius of 2 m, $R_0 = 2$) with an approximately 10-m distance between two neighboring waypoints and then obtained a 130-m-length route in total. The AAC started from the first waypoint and then switched to the go-to-waypoint mission, where it proceeded to the next waypoint along the path once it touched the boundary of the waypoint circle, ending at the eleventh waypoint in numerical order. The simulation test and sea trial were conducted, and the researchers collected the AAC data to demonstrate the AAC's performance with the same configuration.

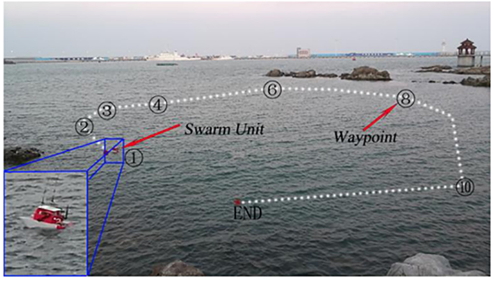


Fig. 9 Sea trials at the China National Deep Sea Center

4.1 Simulation

A simulator named DroneKit-SITL (Rosales et al. 2017) was used to rapidly configure the virtual AAC. This simulator, as shown in Fig. 8, could be installed on the AAC's RPI to test the swarm algorithm without a physical vehicle. Using the simulator with the parameters shown in Table 2, the PI gains of the controller would be easily modified and optimized.

By experience, the higher values of K_P and K_I denote the higher response sensitivity of the controller to the differentiation and its deviation, which are accompanied by a more pronounced overshoot and oscillation. On the other hand, a smaller control parameter value indicates the weaker overshoot and oscillation of the controller and the relatively slower response of the control system to the deviation and its differentiation. Normally, if the overshoot is extremely large, parameter K_P can be slightly reduced with an appropriate increase of parameter K_I. Conversely, if the convergence rate is notably slow, the parameter K_P can be appropriately increased, and the parameter K_I can be reduced slightly to achieve a local optimal control. The appropriate PI gains ($K_P = 0.2$; $K_I = 0.05$) were archived until the desired performance was reached. Then, the simulation tests were repeated for 10 times with the PI gains.

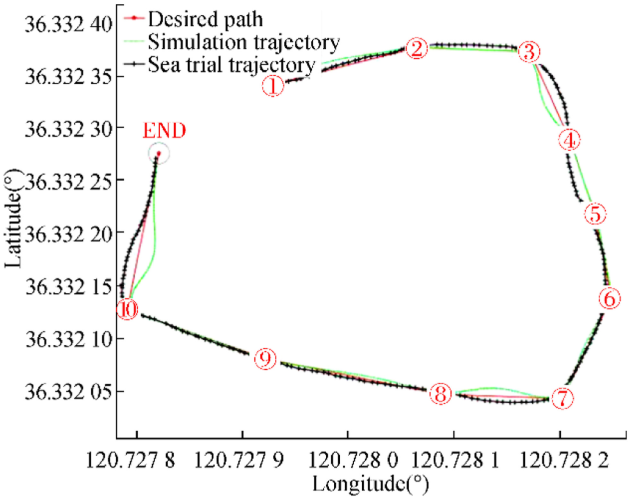


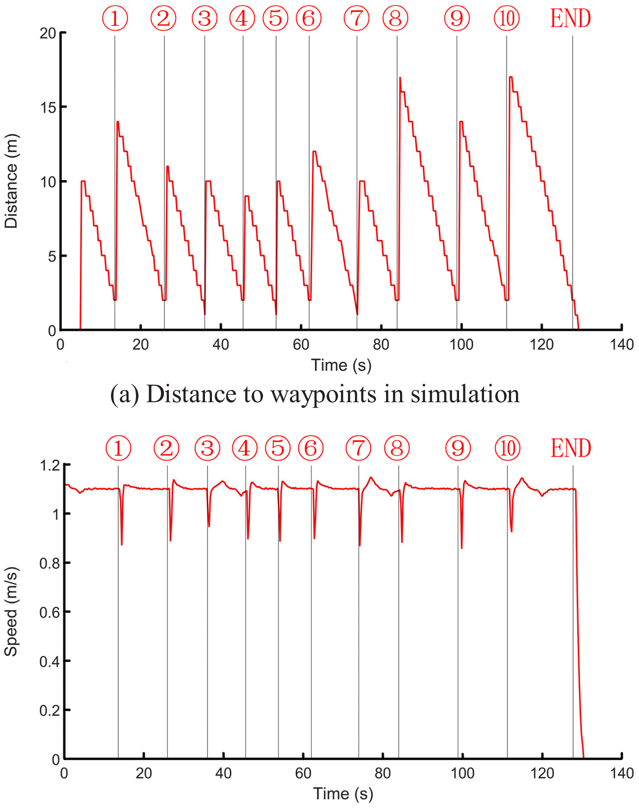
Fig. 10 Random set of simulation and sea trial results

4.2 Sea Trials

Field tests are susceptible to the environmental condition due to the size of the AAC platform. For this reason, tests can only be performed in ideal weather conditions, including steady winds, small waves, and no rain. A series of sea trials at the China National Deep Sea Center is as in an area of 50 × 60 m^2 sea (see Fig. 9) with a wind speed of 0.5 m/s from east to west to demonstrate the AAC's performance. In the experiments, two test runs were conducted for each of the eight AACs, featuring 16 runs in total, with the same PI gains as the simulator.

4.3 Results and Discussion

Figure 10 shows a comparison of the desired path, simulation trajectory, and sea trial trajectory, where one run test datum was randomly selected from 10 time simulations and 16 sea trial runs in random. The AAC passed through



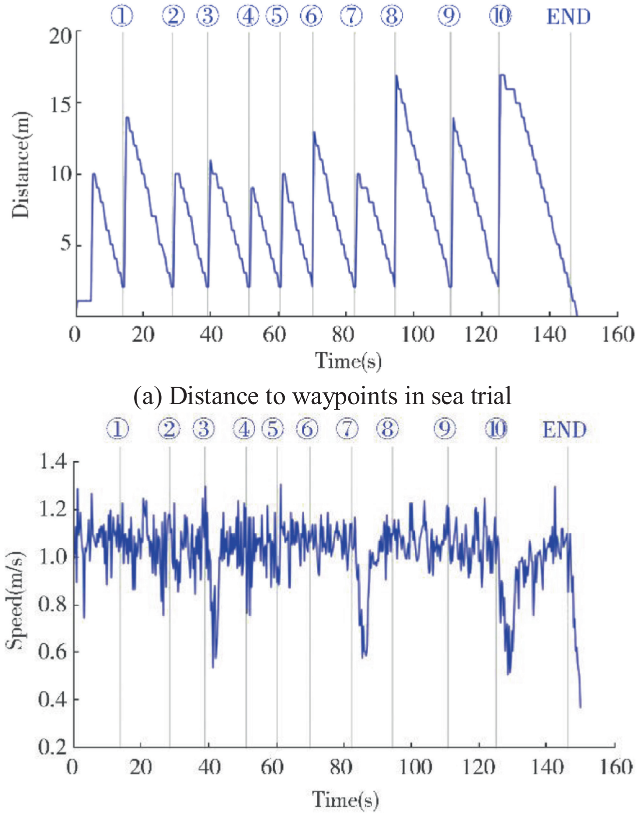
(a) Distance to waypoints in simulation

(b) Speed of AAC in simulation

Fig. 11 Speed controller in simulation (the numbers ① to ⑩ and END denote the time that the AAC reached the boundary of corresponding waypoint circles). **a** Distance to waypoints in simulation. **b** Speed of the AAC in simulation

all the waypoint circles and traced the path within a small offset. The maximum error between the desired path and sea trial trajectory was about 2 m, occurring at the eleventh waypoint, where the simulation offset deviated in the opposite direction. This finding is mainly attributed to the turbulence near the shore, where winds and waves heavily beat the coast in the sea trial. Thus, the operators can set a 2-m-radius collision avoidance safety circle around each AAC for the AACSS in real-world swarm algorithm tests. Afterward, by comparing the 16 sea trial runs, the results showed the acceptable differences among the sea trial trajectories (maximum offset was below 0.5 m).

In the simulation tests, an AAC exceeded the distance from the first waypoint to the eleventh waypoint in 123.8 s on average (Fig. 11). However, all runs lasted about 143 s with a deviation of 3 s to finish the whole course in the sea trial (Fig. 12). The average speed in simulation and sea trial was 1.05 and 0.91 m/s, respectively. The simulated AAC was 19.2 s faster than that operating in the real world. The difference between these two test types can be explained from the speed and yaw speed, as shown in Figs. 12 and 13.



(a) Distance to waypoints in sea trial

(b) Speed of AAC in sea trial

Fig. 12 Speed controller in sea trial (the numbers ① to ⑩ and END denote the time that the AAC reached boundary of corresponding waypoint circles). **a** Distance to waypoints in sea trial. **b** Speed of the AAC in sea trial

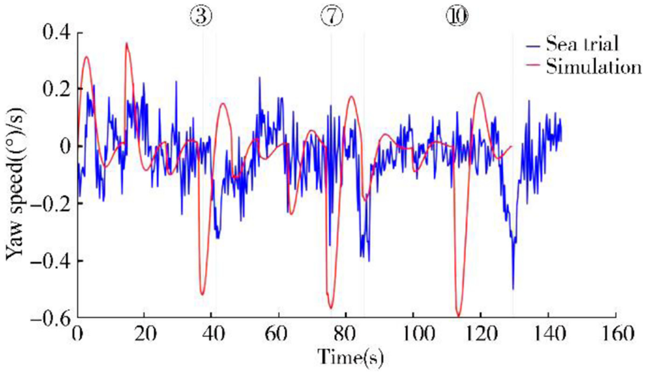


Fig. 13 Yaw speed in simulation and sea trial (the number ③, ⑦, and ⑩ denote the time that the AAC reached boundary of corresponding waypoint circles)

The speed decreased with fluctuations as the AAC's distance to the waypoint reduced in the sea trial, whereas it was displayed smoothly in the simulation where the wind drag and wave were ignored. At the first waypoint, both the simulated and actual AAC could achieve full speed in 3 s, although the simulated AAC responded faster than the one in real world. Then, the AACs ran toward the next 10 waypoints and slowed down after reaching the boundary of the end waypoint. During the whole processing of path following, the real AAC spent more energy and time to deal with the irregular disturbance from winds and waves. For the same reason, no two runs were the same in the sea trials. Another consequence of wind and wave speed is the side slip, with the diversion from the desired path shown in Fig. 10 and yaw speed vibration shown in Fig. 13.

At large corners (③ waypoint, ⑦ waypoint, and ⑩ waypoint), as waves may contribute to add extra resistance to the AAC's yaw speed, the simulation results obtained three larger troughs compared with those in the sea trial (Fig. 13), which appeared with a series of small oscillations. Although a minimal overshot existed in the heading following as shown in Figs. 14 and 15, the heading controller showed good performance in both the simulation and sea trials. The frequency and amplitude of overshot in the sea trial were slightly less than those in the simulation as a result of catamaran's capability for

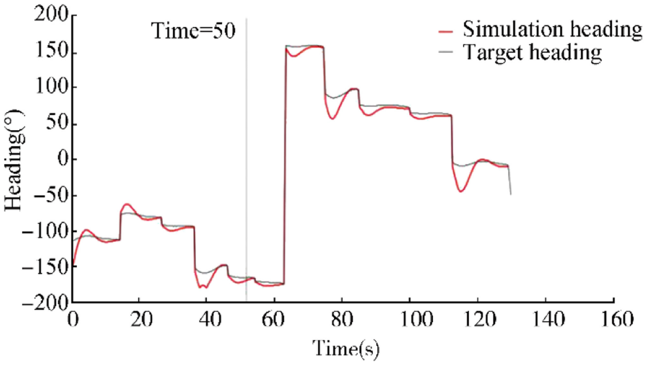


Fig. 14 Heading controller in simulation

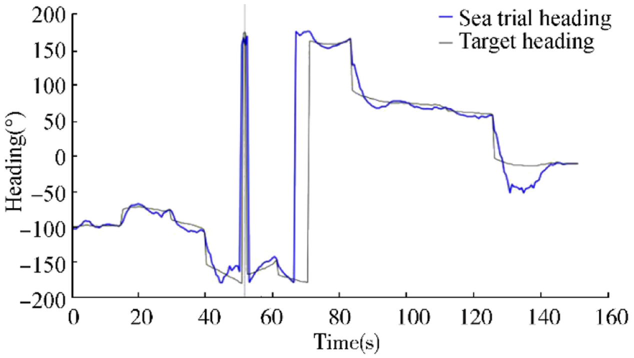


Fig. 15 Heading controller in sea trial

seakeeping in the real world. At 50 s, a peak existed in the sea trial as the heading range is a circle from -π to π in radians.

Considering the limitation of the complex sea condition, the environment model should be further developed to include a simple dynamic weather model, such as those for winds and waves, which feature a high impact on maritime platform (Wang et al. 2015). Winds are unpredictable, and change speed and direction. A simple code where wind speed and direction change based on knowledge base is proposed where an AAC runs on the water surface; this code should be sufficient to tune the PI gains online to achieve better capability for go-to-waypoint and path following. The addition of obstacles could be performed for swarm behaviors by including objects, such as other AACs and stones, above the water surface. The controller would develop the necessary skills to avoid obstacles.

5 Conclusions and Further Work

To date, swarm robotics systems have seldom been applied to tackle real-world applications and are still being simulated or tested in the laboratory. Most studies solely focus on obtaining the desired collective behaviors and understanding their properties. To avoid the problems that arise in real-world applications, this article presented a solution to prepare a simple, inexpensive, flexible, and open platform with a speed controller and heading controller for swarm research on water surfaces. A sea trial with waves and winds successfully demonstrated the capability of go-to-waypoint and path following for a single AAC outside the strictly controlled laboratory conditions. In the future, the researchers plan to develop an AACSS of 7 or more AACs to conduct a hydrological monitoring mission in sea with waves and winds.

References

Ballesteros-Gómez A, Rubio S (2011) Recent advances in environmental analysis. Analytical chemistry 83(12):4579–4613. https://doi.org/10.1021/ac200921j

Brambilla M, Ferrante E, Birattari M, Dorigo M (2013) Swarm robotics: a review from the swarm engineering perspective. Swarm Intelligence 7(1):1–41. https://doi.org/10.1007/s11721-012-0075-2

Castello E, Yamamoto T, Libera FD, Liu W, Winfield AFT, Nakamura Y, Ishiguro H (2016) Adaptive foraging for simulated and real robotic swarms: the dynamical response threshold approach. Swarm Intelligence 10(1):1–31. https://doi.org/10.1007/s11721-015-0117-7

Christensen AL, O'Grady R, Dorigo M (2009) From fireflies to fault tolerant swarms of robots. IEEE Trans Evol Comput 13(4):1–12. https://doi.org/10.1109/TEVC.2009.2017516

Crespi V, Galstyan A, Lerman K (2008) Top-down vs bottom-up methodologies in multi-agent system design. Auton Robot 24(3):303–313. https://doi.org/10.1007/s10514-007-9080-5

Danymol R, Ajitha T, Gandhiraj R (2014) Real-time communication system design using RTL-SDR and Raspberry Pi. International Conference on Advanced Computing and Communication Systems. Coimbatore, India, December 1–5. https://doi.org/10.1109/ICACCS.2013.6938691

Dong X, Zhou Y, Lu G, Zhong Y (2016) Time-varying formation control for unmanned aerial vehicles with switching interaction topologies. Control Eng Pract 46:26–36. https://doi.org/10.1016/j.conengprac.2015.10.001

Duarte M, Gomes J, Costa V, Rodrigues T, Silva F (2016). Application of swarm robotics systems to marine environmental monitoring. *Oceans Conference*. IEEE, Shanghai, China, April 10–13. https://doi.org/10.1109/OCEANSAP.2016.7485429

Dunbabin M, Marques L (2012) Robots for environmental monitoring: significant advancements and applications. IEEE Robot Autom Mag 19(1):24–39. https://doi.org/10.1109/MRA.2011.2181683

Floreano D, Mitri S, Magnenat S, Keller L (2007) Evolutionary conditions for the emergence of communication in robots. Curr Biol 17(6):514–519. https://doi.org/10.1016/j.cub.2007.01.058

Fossen T (1994) Guidance and control of ocean vehicles. Wiley, New York 133–183

Leonard NE, Paley DA, Davis RE, Fratantoni DM, Lekien F, Zhang F (2010) Coordinated control of an underwater glider fleet in an adaptive ocean sampling field experiment in Monterey bay. J Field Robot 27(6):718–740. https://doi.org/10.1002/rob.20366

Li M, Lu K, Zhu H (2008) Robot swarm communication networks: architectures, protocols, and applications. The Third International Conference on Communications and Networking in China, Hangzhou, China, August 162–166. https://doi.org/10.1109/CHINACOM.2008.4684993

Meier L, Tanskanen P, Heng L, Lee GH, Fraundorfer F, Pollefeys M (2012) PIXHAWK: a micro aerial vehicle design for autonomous flight using onboard computer vision. Autonomous Robotics 33(1–2):21–39. https://doi.org/10.1007/s10514-012-9281-4

Parker CA, Zhang H (2011) Biologically inspired collective comparisons by robotic swarms. The International Journal of Robotics Research 30(5):524–535. https://doi.org/10.1177/0278364910397621

Rosales C, Soria C, Carelli R, Rossomando F (2017) Adaptive dynamic control of a quadrotor for trajectory tracking. In 2017 International Conference on Unmanned Aircraft Systems, Miami, USV, June 547–553. https://doi.org/10.1109/ICUAS.2017.7991492

Şahin E (2004) Swarm robotics: From sources of inspiration to domains of application. International Workshop on Swarm Robotics, Berlin, Genman, July 10-20. https://doi.org/10.1007/978-3-540-30552-1_2

Soysal O, Bahç E (2007) Aggregation in swarm robotic systems: evolution and probabilistic control. Turkish Journal of Electrical Engineering & Computer Sciences 15(2):199-225

Valada A, Velagapudi P, Kannan B, Tomaszewski C, Kantor G, Scerri P (2014) Development of a low cost multi-robot autonomous marine surface platform. Field and Service Robotics 92:643–658. https://doi.org/10.1007/978-3-642-40686-7_43

Wang L, Chu XM, Liu CG (2015) Different drive models of USV under the wind and waves disturbances MPC trajectory tracking simulation research. In 2015 International Conference on Transportation Information and Safety, Wuhan, China, June 563–568 https://doi.org/10.1109/ICTIS.2015.7232199

Wolf MT, Rahmani A, de la Croix JP, Woodward G, Vander Hook J, Brown D, Pomerantz M (2017) CARACaS multi-agent maritime autonomy for unmanned surface vehicles in the Swarm II harbor patrol demonstration. In Unmanned Systems Technology XIX, International Society for Optics and Photonics 10195:101950O. https://doi.org/10.1117/12.2262067

Wright RG, Baldauf M (2016) Hydrographic survey in remote regions: Using vessels of opportunity equipped with 3-dimensional forward-looking sonar. Marine Geodesy 39(6): 439–457. https://doi.org/10.1080/01490419.2016.1245226

Xu G, Shen W, Wang X (2014) Applications of wireless sensor networks in marine environment monitoring: A survey. Sensors 14(9):16932–16954. https://doi.org/10.3390/s140916932

Yu Y, Chen X, Lu Z, Li F, Zhang B (2017) Obstacle avoidance behavior of swarm robots based on aggregation and disaggregation method. Simulation Transactions of the Society for Modeling & Simulation International 93(11):885-898. https://doi.org/10.1177/0037549717711281

Zhu B, Xie L, Han D, Meng X, & Teo R (2017) A survey on recent progress in control of swarm systems. Science China Information Sciences 60(7):070201. https://doi.org/10.1007/s11432-016-9088-2