Neural Swarm Control Algorithm for Underwater Vehicles

TOMASZ PRACZYK, PIOTR SZYMAK
IT Department
Polish Naval Academy
Śmidowicza 69, 81-127, Gdynia
POLAND

Abstract: The paper presents the application of an evolutionary recurrent neural network to control the swarm of underwater vehicles. In the swarm, one vehicle is the leader and the others are followers. The leader leads the swarm along a predefined trajectory without regard for the followers while the followers follow the leader and avoid collisions with all other vehicles. Avoiding collisions by the swarm with external obstacles is done by changing the depth. The leader is responsible for detecting the obstacles and informing all the followers about the need to change the depth. To follow the leader, the followers use the information about the distance to it. Directional information is unavailable to them. To avoid collisions inside the swarm, the followers use short-range sensors.

Key-Words: neural networks, swarm, autonomous underwater vehicles, evolutionary computation, control

Received: December 8, 2022. Revised: August 16, 2023. Accepted: September 22, 2023. Published: October 17, 2023.

1 Introduction

Underwater vehicles are robots that allow the implementation of many different tasks in the underwater environment. We distinguish remotely operated vehicles (ROV), i.e. unmanned vehicles that are controlled by a human-operator and connected to him/her using a special cable transmitting control signals and often also energy, and autonomous vehicles (AUV) that can operate without human assistance.

Autonomous vehicles can perform their tasks not only independently but also in larger teams or swarms. The team of vehicles is understood in the paper as loosely cooperating vehicles, each of which can also operate independently of the others and is equipped with a global underwater dead reckoning navigation system for this purpose. Providing vehicles operating in teams with the ability to operate independently makes these vehicles expensive and often very complicated to operate.

Another solution is a swarm of vehicles understood in the paper as a group of vehicles closely cooperating and moving in a compact formation. If we assume that one of these vehicles, say leader, is responsible for global navigation and guiding the entire swarm along a predefined path, then the remaining vehicles, say followers, can be relieved of the need for expensive devices for long-range underwater navigation and thus become cheaper, smaller and easier to use.

However, for followers to be able to follow the leader and at the same time avoid obstacles and neighbors, an appropriate control system is necessary. The proposed system is a recurrent neural network (RNN) trained using a neuro-evolutionary algorithm called Hill Climb Assembler Encoding (HCAE) [1].

The network is supplied with two types of information, i.e. information about the distance from the leader which is provided by the leader itself via an acoustic communication channel, and information about nearby objects which is provided by vehicle sensors. Leader-related directional information is unavailable to followers which means that they know how far away the leader is, but they have no idea in what direction they can expect it, which is a serious difficulty for the neural control system. What is more, information about the distance is provided rarely. It is delivered to the followers one by one, which means that the more vehicles are in the swarm the less often each of them receives information about the distance from the leader.

Information about nearby objects comes from sensors and is obtained much more often than distance information. The problem, in this case, is that the followers do not know the nature of the observed object. They do not know if it is another follower, a leader, or an obstacle. To facilitate the task of the followers, it was assumed that the leader is responsible for detecting obstacles and avoiding them consists in changing the depth - vehicles pass over the obstacle. However, for the leader to be able to detect obstacles, none of the followers must be in the field of view of the leader's sensors. What is more, the lack of danger of collision with obstacles (at least in theory) does not mean that followers do not see them. The swarm can move close to obstacles, with the effect that the sensors of the followers can still detect them. As a consequence, followers have no fear of obstacles, but they still can notice them and cannot distinguish them from swarm vehicles.

To verify the proposed swarm control system, it was tested in simulation conditions. Training of the system took place on GPU servers, while the visualization of swarm behavior was in MOOS IvP environment [2]. Both the leader and followers were the same, i.e. they behaved according to the same kinematic model. The swarm consisted of one leader and four followers. The leader moved along a predefined trajectory that had both straight sections and turns. During the voyage, the leader did not pay attention to other vehicles which means that in-swarm collision avoidance was the task of the followers. The tests were performed with and without external obstacles.

The contribution of the paper is as follows:

1. Neural swarm control system for follower vehicles is proposed,
2. The characteristic features of the system are: (i) followers have to follow the leader based only on distance information, directional information is unavailable, (ii) distance information is rare, (iii) followers cannot be in the field of view of the leader sensors, (iv) follower sensors cannot distinguish external objects from in-swarm objects,
3. The system was verified in simulation conditions.

The rest of the paper is as follows: section two outlines related work, section three details the proposed system, section four reports experiments, and the final section concludes the paper.

2 Related Work

The subject of robot swarm technology is increasingly undertaken in world literature by both scientists and practitioners. Papers in this field, however, mainly apply to air, land, and sometimes surface robots. Due to the complexity of the underwater environment and the problems that the environment generates, papers presenting examples of underwater swarms are rarer than those referring to the aforementioned air and land swarms.

Interesting examples of underwater swarms are given in works [3], [4]. They present aggregation, dispersion, and diffusion swarm behaviors. Underwater robots concentrate in one place for some purpose or fill some space as much as possible.

The fountain maneuver and circle formation are presented in [5], [6], [7]. In this case, the task of pray vehicles is to evade predators.

Hunting swarm behavior is presented in [8], [9], [10], [11]. In this case, the task of the vehicles is to reach a target vehicle, and then to form an encircling formation around the target.

Leader-following swarm strategy with a fixed formation is demonstrated in [12], [13], [14]. A special case of the leader-following approach is the application of consensus algorithms [15], [16], [17]. The task of the algorithm is to achieve a consensus on a common goal in a group of cooperating robots.

3 Swarm control system

The task of the swarm control system is to lead the followers to a certain endpoint along a predetermined trajectory. The followers, according to the assumption, are low-cost vehicles that cannot independently move over long distances. As a consequence, to reach the destination point, they must follow another vehicle (leader) that shows them the way. To follow the leader, the followers are fed with information about the distance from the leader. However, this information is provided relatively rarely - the more vehicles in the swarm, the less often this information reaches the followers.

To reach their destination, in addition to following the leader, the followers must avoid other vehicles in the swarm. To observe the surroundings, the followers use two different sensors, i.e. cameras placed on the sides and rear of the vehicles, and sonar looking forward. The cameras have a maximum range of 5m, while the sonar range is 30 meters. The angular observations range of the camera and sonar is shown in Figure 1.

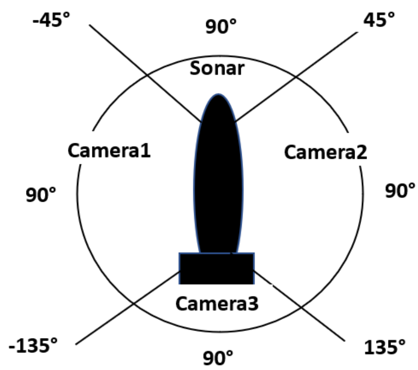

Figure 1: Follower observation sectors

It is assumed that external obstacles are avoided by changing the depth and the decision to avoid obstacles is made by the leader which sends a new depth for all vehicles. For the leader to be able to detect obstacles and inform followers about them, none of the followers can enter the leader's field of vision. Otherwise, every follower noticed by the leader would be interpreted as an obstacle to be avoided.

The above solution to the problem of avoiding obstacles by the whole swarm results from the inability of the vehicles to distinguish obstacles from other vehicles. Using sonar as the main long-range sensor, it is very difficult to determine what object you are dealing with. This makes it difficult for followers to adjust the appropriate behavior strategy - when the object is not dangerous because it moves in the same direction and when the object is a threat because it does not move or moves in the opposite direction. Moreover, the echo seen in the sonar image may correspond to several objects located at the same distance, but at different depths.

Due to the difficulties in interpreting the sonar image, it was finally decided on a solution in which only the leader detects obstacles and whatever it detects is an obstacle that must be avoided. In turn, the followers are only responsible for avoiding other followers and the leader. In this way, responsibility is divided and each vehicle knows what it is dealing with in the sonar image.

Since it is assumed that the leader deals only with obstacles and moves along a predefined trajectory to an endpoint, the task of keeping the swarm in its entirety is the task of the followers only. To this end, each follower is equipped with a recurrent neural network that is fed with the following information:

1. Distance to the nearest object in sonar observation sector scaled to the range <0,1>,

2. Distances to the nearest objects in three camera observation sectors scaled to the range <0,1>,

3. $\Delta D_t^M = |D_t^L - D^D| - |D_{t-1}^L - D^D|$, where D_t^L is the distance to the leader received by the follower at time t and D^D is a desired distance to the leader. ΔD^M provides the network with information about whether the distance error increases ($\Delta D^M > 0$) or decreases ($\Delta D^M < 0$) in the period between two leader messages, and how big is the change. It is a counterpart of the rate of change of error with respect to time applied in PID controller.

4. $D_{<0,1>,t}^S = \frac{D_t^S}{T_{max}^{D^L}}$, where $D_t^S = \begin{cases} T_{max}^{D^L} & \sum_{k=1}^t \Delta D_k^M > T_{max}^{D^L} \\ -T_{max}^{D^L} & \sum_{k=1}^t \Delta D_k^M < -T_{max}^{D^L}, \\ \sum_{k=1}^t \Delta D_k^M & \text{otherwise} \end{cases}$
and $T_{max}^{D^L}$ is the maximum acceptable error of the distance to the leader. $D_{<0,1>}^S$ accumulates the changes of distance errors over time. It can be negative if the distance error is positive at the very beginning of the swarm operation. It is equivalent to the sum of errors in a PID controller.

The task of the network is to determine the heading and speed of the vehicle by the following formulas:

1. $H_{t+1}^D = A_{360}(H_t) + 180O_t^H$, where H^D is a desired heading, H is actual heading, A_{360} is a function which converts input angle to the range $< 0, 360)$, and O_H, is output of the network corresponding to heading. In consequence, the task of the network is not to determine the heading but the change of the heading.

2. $V_{t+1}^D = V_{max}O_t^V$, where V^D is a desired speed, V_{max} is maximum speed of the follower, and O_V is output of the network corresponding to speed.

The network does not specify the depth at which the vehicle is to move. As already mentioned, the depth at which each follower moves depends on the leader. The leader sends the depth through the acoustic channel if it needs to be changed. The depth change is handled by the low-level control system.

The neural network is trained using a neuro-evolutionary algorithm called Hill Climb Assembler Encoding [1]. It is an algorithm that represents a network in the form of a matrix and constructs this matrix in many successive evolutionary iterations. Unlike other evolutionary algorithms, the genotype does not represent the entire network (matrix) but some part of it. As a consequence, the network is encoded not in one genotype but in a sequence of genotypes coming from other runs of the evolutionary process. The operation of HCAE can be compared to a situation in which the object subject to optimization is improved little by little, piece by piece. Each piece is the result of a different evolutionary process. The detailed specification of HCAE is given in [1].

4 Experiments

To verify the effectiveness of the swarm control system described above, simulation tests were carried out. During the tests, the task of the swarm consisting of a leader and four followers was to move along a predefined trajectory of 2000 meters. Two trajectories were used to train the neural networks, and the other two trajectories were used to verify the effectiveness of the system. Each of the trajectories had both straight sections and turns to the right and left.

In the first phase of testing, the vehicles had no obstacles to deal with, while in the second phase, round obstacles with a diameter of 20 meters were placed along the route of the vehicles. Since it was assumed that the leader can correctly detect obstacles and properly control the depth of the swarm, all obstacles were

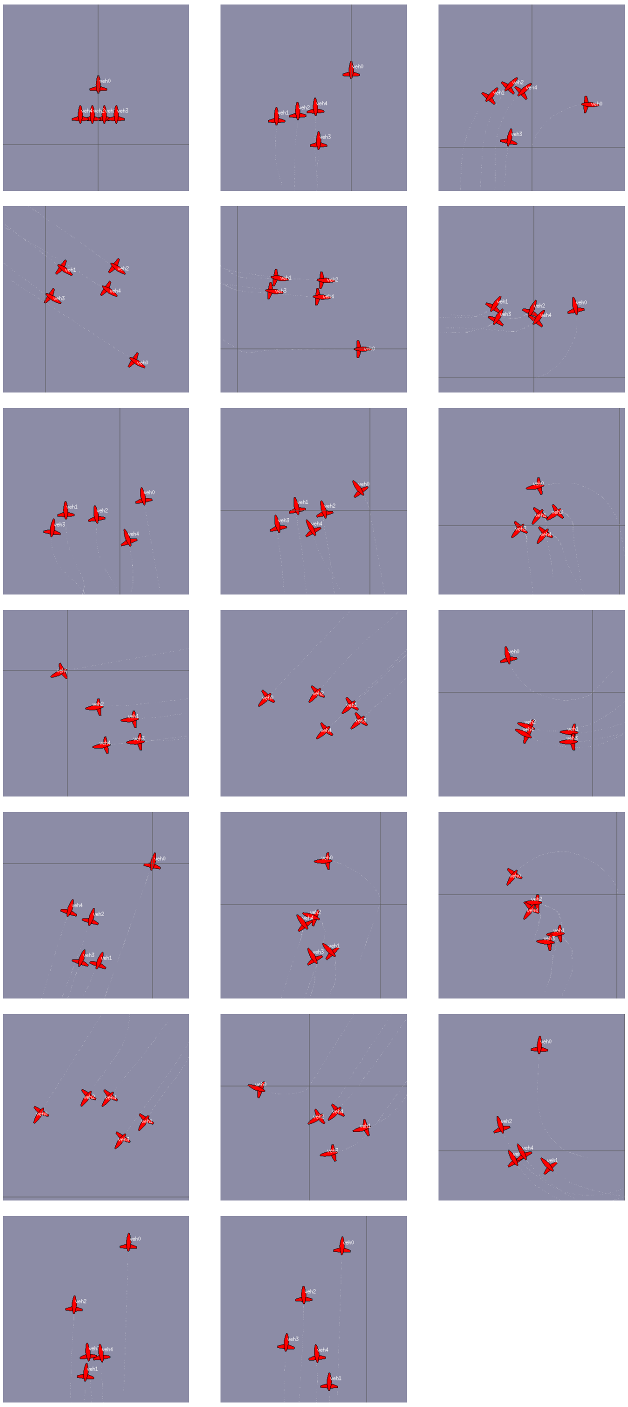

Figure 2: Example behavior of the swarm in the scenario without obstacles (time in the images flows first from left to right and then from top to bottom). The visualization of vehicle behavior was carried out using the alogview MOOS application [2]

located under the trajectory of the swarm. As a consequence, obstacles did not cause collisions, but if they were within the range of the sensors, they were noticeable by the control system.

All vehicles behaved by the kinematic model implemented in the MOOS uSimMarine application [2]. The HCAE algorithm was used to train the networks and the following fitness function was applied to evaluate the generated neural solutions: $F(network)_D = N^I + \frac{1}{1+E^D_{max}}$, where N^I is the number of simulation steps (see further), and E^D_{max} is the maximum error of the distance to the leader among all followers.

The number of steps in evaluating each of the evolved networks (N^I) depended on their effectiveness. The evaluation was interrupted in four situations. Firstly, when $E^D_{max} > T^{D^L}_{max}$, i.e. if any follower was further from the leader than the assumed threshold $T^{D^L}_{max}$. Secondly, when any follower entered the leader's field of vision. Thirdly, when all followers made the same decisions regarding the direction of movement and speed for a short period. Fourthly, when vehicles collided, i.e. the distance between any pair of vehicles was less than 2 meters.

The more important parameters of the simulations are as follows: the number of followers=4, $D^D = 30$m, $T^{D^L}_{max} = 40$m, $V_{max} = 2$m/s, the time interval between successive sensor data=1s, the time interval between successive messages containing information about the distance to the leader=4s (as already mentioned, information about the distance was received by the followers one by one, it was sent by the leader to each follower every 1s, first to follower no. 1, then to follower no. 2, and so on), simulation time step=0.1s, the number of input neurons in the network=6 (four inputs for observation sectors, one input for ΔD^M_t, and one input for $D^S_{<0,1>,t}$), the number of output neurons in the network=2 (one for heading and one for speed), the number of hidden neurons=12.

In the first phase of testing, when the vehicles did not have to deal with obstacles, initially there were no restrictions on the turning speed of the leader. This speed depended only on the leader's maneuverability. It turned out, however, that in the absence of restrictions on the behavior of the leader, the followers were unable to follow the leader without entering the field of vision of its sonar.

To avoid the above problems, the turning maneuver of the leader was slowed down in such a way that at a distance of 10 meters, the leader could only change course by 30 degrees. After using this solution, it turned out that the followers can effectively follow the leader and not enter the field of vision of its sensors. The vast majority of neural networks that were constructed during the learning process were able to control the followers along both test trajectories, from the start point to the endpoint. Example behavior of the swarm is depicted in Figure 2.

Regardless of the neural network, the behavior of followers was very similar. Very often they moved

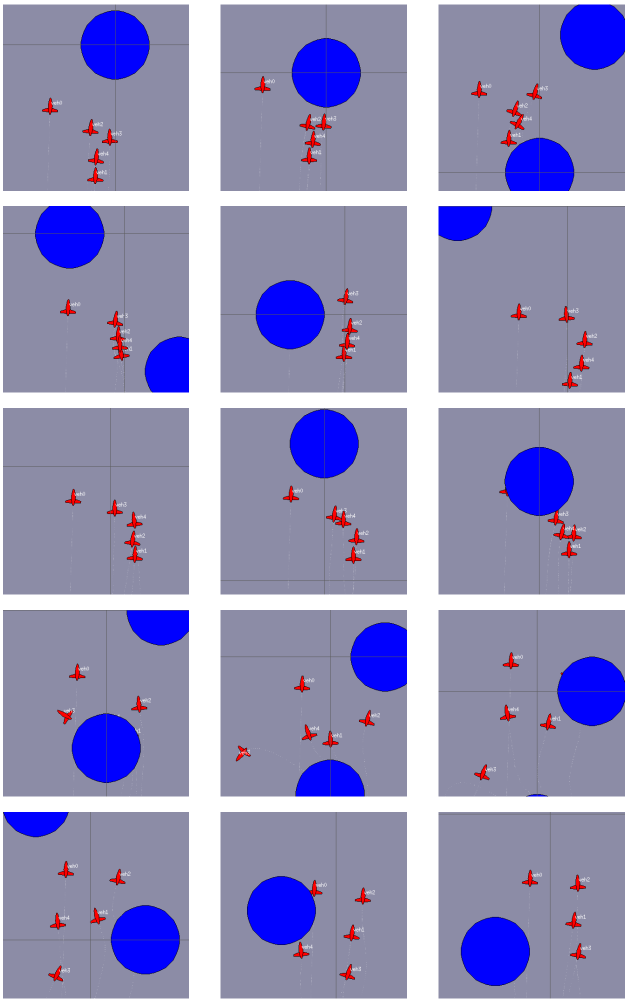

Figure 3: Example behavior of the swarm in the scenario with obstacles (if the vehicles are covered by an obstacle, it means that they are over the obstacle - in the alogview MOOS application used to visualize the simulation results, all additional objects are always presented in the foreground in relation to the vehicles.)

close to each other, observing the behavior of neighbors with the help of sensors. This behavior could help followers to follow the leader even when information about the distance from the leader is very rare. By observing the neighbors, each follower obtains indirect information about the behavior of the leader. A change in the direction of movement by a neighbor may be due to the change in the direction of the leader's movement and the message about the distance received a moment ago by the neighbor. Although only one follower at a time receives a distance message that allows it to adjust its behavior to follow the leader, other vehicles in its vicinity, seeing the change in the neighbor's behavior, can do the same even though they have no information about the leader's location.

The behavior that allows the followers to avoid the field of vision of the leader is to reduce speed or even stop when the leader is making a turn toward the followers. This behavior can be seen, for example, in sub-figures (3,2), (3,3), (2,5) in matrix-Figure 2.

In the second phase of the tests, obstacles were introduced into the environment. They had a circular shape, and a diameter of 20 meters and were placed at the end of the leader's trajectory, i.e. on a straight long section leading to the endpoint. During the simulation, it was assumed that the swarm moves above obstacles that are detected by the leader - each time the leader detects an obstacle, it orders a change of depth to avoid obstacles by moving over them. This approach means that the followers did not have to avoid obstacles. All they had to do was to follow the leader in a situation where, in addition to other vehicles, obstacles indistinguishable from vehicles were also visible in the sonar image. The exemplary behavior of a swarm in the presence of obstacles is presented in Figure 3.

The addition of obstacles on the trajectory of vehicles introduced difficulties in the evolution of effective neural networks, i.e. those that were able to lead a swarm from the starting point to the endpoint of the trajectory. In contrast to the test scenario without obstacles, the number of effective neural networks for the scenario with obstacles decreased significantly. In this case, only about 10% of the runs of the evolutionary process resulted in the generation of fully effective neural networks.

As for the reaction of followers to obstacles, it occurred only when vehicles passed over obstacles. Obstacles on the side of the followers did not cause any disturbances in proper movement behind the leader. They were usually at a safe distance from the vehicles and consequently were not considered a threat that required a response. On the other hand, obstacles located directly under the followers generated sonar echoes in close proximity to the vehicles. In this case, the reaction was mostly a slight change of course, as if the obstacles were gently pushing the followers away. There were also cases when followers considered obstacles as an imminent collision hazard and performed an evasive maneuver consisting of making a full circle to the left or right - see sub-figures (1,4), (2,4), (3,4) in matrix-Figure 3. After performing this maneuver, the followers returned to the swarm.

5 Conclusions

The paper presents the use of recurrent neural networks constructed in an evolutionary way to control underwater vehicles acting as followers in a swarm consisting of one leader and a group of followers. The only task of the followers is to follow the leader while avoiding collisions with neighboring vehicles. No specific formation of followers is required.

During the experiments, the results of which are presented in the paper, the vehicles were unable to distinguish obstacles from other vehicles in the swarm using the only long-range sensor which was sonar. In consequence, it was decided that avoiding collisions with obstacles external to the swarm is the responsibility of the leader which detects obstacles and, in the event of a threat, decides to change the depth for the entire swarm. However, for the leader to be able to detect obstacles and not confuse them with followers, none of the followers must be in the field of vision of the leader sonar. Such a limitation was a serious challenge for the neural control system, especially on bends.

If we assume that the leader is able to quickly detect obstacles and change the immersion depth of the swarm, then we can conclude that the followers are not at risk of colliding with external objects. However, if followers move near obstacles, they are still visible to them, which makes it difficult to follow the leader because it is not known whether the detected object is another vehicle or an obstacle. The impossibility of correctly interpreting information from the sonar is another problem that the neural control system had to face.

An additional difficulty for the system was also the limited information available for decision-making. To follow the leader, the followers were supplied with information about the distance to it. However, this information was provided rarely, one by one to each follower. Directional information was not available to the followers. In addition, the followers used limited-range sensors such as sonar and cameras to avoid collisions.

Despite all the challenges that the neural control system had to face, the simulations, the results of which are presented in the paper, showed that the use of recurrent neural networks as a high-level control system of the followers, i.e. the system determining the direction of movement and speed of vehicles, allows for collective movement of vehicles along the route designated by the leader. As it turned out, vehicles equipped with a neural network can follow the leader both in an ideal, obstacle-free marine environment where every nearby object is a threat - it is just another vehicle in a swarm that can lead to a collision and in an environment containing obstacles in which followers have to deal with both other vehicles that may pose a threat and with underwater objects that do not pose a threat because they are at a different depth away from the swarm.

References:

[1] T. Praczyk, Hill Climb Assembler Encoding: Evolution of small/mid-scale artificial neural networks for classification and control problems, *Electronics*, Vol. 11, No. 13, 2022, doi:10.3390/electronics11132104.

[2] Module page; https://oceanai.mit.edu/moos-ivp/pmwiki/pmwiki.php?n=Main.HomePage (27/4/23)

[3] M. Bodi, C. Moslinger, R. Thenius, T. Schmickl, Beeclust used for exploration tasks in autonomous underwater vehicles, *IFAC-PapersOnLine, 8th Vienna International Conference on Mathematical Modelling*, Vol. 48, No. 1, 2015, pp. 819–824, doi:https://doi.org/10.1016/j.ifacol.2015.05

[4] E. Petritoli, M. Cagnetti, F. Leccese, Simulation of autonomous underwater vehicles (auvs) swarm diffusion, *Sensors*, Vol. 20, No. 17, 2020, doi:10.3390/s20174950.

[5] F. Berlinger, P. Wulkop, R. Nagpal, Self-organized evasive fountain maneuvers with a bioinspired underwater robot collective, *in: 2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9204-9211

[6] F. Berlinger, M. Gauci, R. Nagpal, Implicit coordination for 3d underwater collective behaviors in a fish-inspired robot swarm, *Science Robotics*, Vol. 6, No. 50, 2021

[7] F. Berlinger, *Blueswarm: 3d self-organization in a fish-inspired robot swarm*, Ph.D. thesis, Harvard University Graduate School of Arts and Sciences, 2021

[8] M. Chen, D. Zhu, A novel cooperative hunting algorithm for inhomogeneous multiple autonomous underwater vehicles, *IEEE Access*, Vol. 6, 2018, pp. 7818-7828, doi:10.1109/ACCESS.2018.2801857

[9] H. Liang, Y. Fu, F. Kang, J. Gao, N. Qiang, A behavior-driven coordination control framework for target hunting by uuv intelligent swarm, *IEEE Access*, Vol. 8, 2020, pp. 4838-4859, doi:10.1109/ACCESS.2019.2962728

[10] L. Cai, Q. Sun, Multiautonomous underwater vehicle consistent collaborative hunting method based on generative adversarial network, *International Journal of Advanced Robotic Systems*, Vol. 17, No. 3, 2020, 1729881420925233, arXiv:https://doi.org/10.1177/1729881420925233, doi:10.1177/1729881420925233

[11] Z. Zhao, Q. Hu, H. Feng, X. Feng, W. Su, A cooperative hunting method for multi-auv swarm in underwater weak information environment with obstacles, *Journal of Marine Science and Engineering*, Vol. 10, No. 9, 2022, doi:10.3390/jmse10091266

[12] Y. Zhang, S. Wang, K. M. Heinrich, X. Wang, M. Dorigo, *3d formation control of an underwater robot swarm: Switching topologies, disconnections, and hybrid localization*, Technical Report No. TR/IRIDIA/2020-006, 2021

[13] Zhang, S. Wang, M. K. Heinrich, X. Wang, M. Dorigo, 3d hybrid formation control of an underwater robot swarm: Switching topologies, unmeasurable velocities, and system constraints, *ISA Transactions*, 2022, doi:https://doi.org/10.1016/j.isatra.2022.11.014

[14] L. Li, Y. Li, Y. Zhang, G. Xu, J. Zeng, X. Feng, Formation control of multiple autonomous underwater vehicles under communication delay, packet discreteness and dropout, *Journal of Marine Science and Engineering*, Vol. 10, No. 7, 2022

[15] Z. Yan, D. Xu, T. Chen, W. Zhang, Y. Liu, Leader-follower formation control of uuvs with model uncertainties, current disturbances, and unstable communication, *Sensors*, Vol. 18, No. 2, 2018, doi:10.3390/s1802066

[16] Z. Yan, Y. Wu, X. Du, J. Li, Limited communication consensus control of leader-following multi-uuvs in a swarm system under multiindependent switching topologies and time delay, *IEEE Access*, Vol. 6, 2018, pp. 33183-33200, doi:10.1109/ACCESS.2018.2844817

[17] T. Yang, S. Yu, Y. Yan, Formation control of multiple underwater vehicles subject to communication faults and uncertainties, *Applied Ocean Research*, Vol. 82, 2019, pp. 109-116, doi:https://doi.org/10.1016/j.apor.2018.10.024.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Tomasz Praczyk is the author of the swarm control system, he conducted the system learning and simulations. Piotr Szymak is the author of the text.

Sources of funding for research presented in a scientific article or scientific article itself

The paper is supported by European Defence Agency Project No. B-746-ESM1-GP, entitled "Swarm of Biomimetic Underwater Vehicles II (SABUVIS II)".

Conflict of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.