Distributed Adaptive Neural Network Control Applied to a Formation Tracking of a Group of Low-Cost Underwater Drones in Hazardous Environments

Hoang Anh Pham, Thierry Soriano, van Hien Ngo, Valentin Gies

▶ **To cite this version:**

Hoang Anh Pham, Thierry Soriano, van Hien Ngo, Valentin Gies. Distributed Adaptive Neural Network Control Applied to a Formation Tracking of a Group of Low-Cost Underwater Drones in Hazardous Environments. Applied Sciences, 2020, 10 (5), pp.1732. ⟨10.3390/app10051732⟩. ⟨hal-02883612⟩

HAL Id: hal-02883612

https://hal.science/hal-02883612v1

Submitted on 30 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

applied sciences

MDPI

Article

Distributed Adaptive Neural Network Control Applied to a Formation Tracking of a Group of Low-Cost Underwater Drones in Hazardous Environments

Hoang Anh Pham [1], Thierry Soriano [1,*], Van Hien Ngo [2] and Valentin Gies [3]

[1] COSMER Laboratory, University of Toulon, 83130 Toulon, France; hoang-anh.pham@univ-tln.fr
[2] Hanoi University of Science and Technology, Hanoi 100803, Vietnam; hien.ngovan@hust.edu.vn
[3] IM2NP Laboratory, University of Toulon, 83130 Toulon, France; valentin.gies@univ-tln.fr
* Correspondence: thierry.soriano@univ-tln.fr

Received: 31 January 2020; Accepted: 27 February 2020; Published: 3 March 2020

Featured Application: These applications are suitable for missions, such as oceanic surveillance, undersea oil detection, submarine pipeline monitoring, and seabed explorations.

Abstract: This paper addresses a formation tracking problem of multiple low-cost underwater drones by implementing distributed adaptive neural network control (DANNC). It is based on a leader-follower architecture to operate in hazardous environments. First, unknown parameters of underwater vehicle dynamics, which are important requirements for real-world applications, are approximated by a neural network using a radial basis function. More specifically, those parameters are only calculated by local information, which can be obtained by an on-board camera without using an external positioning system. Secondly, a potential function is employed to ensure there is no collision between the underwater drones. We then propose a desired configuration of a group of unmanned underwater vehicles (UUVs) as a time-variant function so that they can quickly change their shape between them to facilitate the crossing in a narrow area. Finally, three UUVs, based on a robot operating system (ROS) platform, are used to emphasize the realistic low-cost aspect of underwater drones. The proposed approach is validated by evaluating in different experimental scenarios.

Keywords: low-cost underwater robotics; distributed adaptive neural network control; collision and obstacle avoidance; robot operating system (ROS); Gazebo

1. Introduction

In recent years, the application of unmanned underwater vehicles (UUVs) has grown steadily. Several activities related to the offshore industry, such as oceanic surveillance, undersea oil detections, seabed explorations, and so on, have been performed by using a group of low-cost, unmanned, small underwater vehicles. Its advantages are it is an effective, economic, and efficient solution, compared to a single expensive underwater vehicle. An essential topic in the cooperative control of multi-UUVs is the formation tracking problem [1–3]. It requires that all the UUVs reach into a formation with a desired shape from an arbitrary initial pose while the centroid of formation tracks a reference trajectory.

According to [1], there are four main approaches, such as leader-follower approach, behavior-based approach, artificial potential approach, and virtual structure approach, to achieve the formation tracking problem. Moreover, the formation tracking problems for a group of multi-agent systems with nonlinear dynamics and external disturbances have been studied in [1,3–8]. In [1],

the robust distributed formation design for multiple UUVs is presented, where the dynamics of underwater vehicles, which is subject to nonlinearity, parametric uncertainties, and external disturbances, are analyzed by combining the feedback linearization method and the robust compensation theory. Distributed formation tracking control and learning/identification of nonlinear uncertain AUV dynamics by using two-layer, namely, an upper-layer distributed adaptive observer and a lower-layer decentralized deterministic learning controller, is introduced in [3]. This article also addresses some very interesting issues, such as heterogeneous nonlinear uncertain dynamics, local information, and formation control performance. In [4,9], a multi-layer neural network and an adaptive robust controller were developed to overcome unmodeled dynamics of under-actuated autonomous underwater vehicles and external disturbances with the prescribed performance. In addition, in [6], the trajectory tracking problem for a fully actuated autonomous underwater vehicle (AUV) that moves in the horizontal plane was developed. By using two neural networks, the long-time performance of a design control and the unknown dynamics are evaluated and compensated, respectively. For the issues of local information, the MORPH project [10] also introduced a formation control of underwater vehicles, using either inter-vehicle relative position measurements or range-only measurements. In [8], the distributed consensus tracking control problem for multiple strict-feedback systems with unknown nonlinearities under a directed graph topology is presented. The neural network is used to approximate and compensate for unknown nonlinear terms, which aims to design a local consensus controller. In [11], the leader-following consensus problem for a class of second-order multi-agent systems with input quantized is studied. A novel adaptive dynamic quantizer that can effectively reduce quantization error, with dynamic quantization interval and limited quantization level, is proposed.

Furthermore, there are also studies of the tracking control problem for a team of quad-rotors under directed switching topologies [12] and for multiple unmanned helicopters under finite-time conditions [13]. Alternatively, in [14], a distributed consensus-based formation control for nonholonomic wheeled mobile robots is studied. This article addresses the issue, namely, perfect velocity tracking, i.e., the actual control inputs of the robot are equal to the desired control inputs. By designing neural network torque controllers with on-line learning, the robust velocity tracking and the desired formation is achieved and guaranteed. In [15], the input quantization, actuator faults, and unknown nonlinear of a consensus tracking control of multi-agent systems are studied under directed communication topology. This work has introduced smooth functions to compensate for the effect of quantization and bounded stuck faults. In [16], the formation tracking problem was designed relying only on the agents' local coordinate systems so that the centroid of the controlled formation tracks a given trajectory. In [17], a distributed model reference adaptive control (MRAC) design framework is proposed for containment control of heterogeneous general linear multi-agent systems. The main advantage of the model reference adaptive control (MRAC) is that adaptation takes place only when the system does not follow the reference model. This is particularly the case when there are uncertainties in the system under consideration. With respect to the distributed formation tracking control of multiple UUVs under a leader-follower architecture, the reference model is considered as the leading UUV. The remaining vehicles are considered as follow vehicles.

For large-scale oceanic missions in hazardous environments, collision and obstacle avoidance have attracted much attention in recent years. In [18], a stable formation control law and collision avoidance have been investigated for multiple robots with single integrator dynamics under an undirected and connected graph. Formation control and collision avoidance are also presented in [19] for double-integrator systems, which is based on position estimation. In [20], the problem of formation control with collision avoidance for networked Lagrangian systems with uncertain parameters is investigated under directed network topology.

In order to make a simulation of underwater drones, an open source tool that simulates the impact of communications in underwater robotics, namely, UWSim-NET [21], is presented. This is a new extension of UWSim [22], which is developed in the robot operation system (ROS). In [23], an extension

of the open-source robotics simulator Gazebo to underwater scenarios is introduced, namely, Unmanned Underwater Vehicle Simulator (UUV-Simulator). It can simulate multiple underwater robots, and simultaneously allows modeling the underwater hydrostatic and hydrodynamic effects, thrusters, sensors, and external disturbances. In [24,25], the SWARMs middleware architecture is presented for cooperative autonomous underwater vehicles at the mission level.

Motivated by the above discussions, this paper has proposed and implemented a control framework for multiple low-cost underwater drones (i.e., BlueROV-1). Compared to the previous relative studies on the control framework of UUVs for formation tracking problem [1–8], the proposed control framework can allow a group of UUVs to handle simultaneously and continuously the formation tracking problem and collision-obstacle avoidance. Compared with the collision-obstacle avoidance in [18,19], which were studied for single and double integrator systems, the proposed control framework is subject to nonlinearity, parametric uncertainties of UUVs. More specifically, this proposed control framework is a combination of four terms: the consensus control, the neural network control, the robust control, the collision-obstacle avoidance control. Firstly, the consensus term is used to set up a desired formation. Secondly, the neural network control is employed to compensate for the unknown uncertainties. Thirdly, the robust control is presented, which is aimed to achieve asymptotic convergence of the tracking error. Finally, the collision and obstacle avoidance are mentioned to ensure that a group of BlueROV-1 can operate in hazardous environments. Specially, we have also implemented the control framework that uses only relative positions between underwater drones. This is reasonable in real terms that are only equipped with low-cost sensors (i.e., an inertial measurement unit sensor and a camera) as well as limitations of underwater communication. By emphasizing the realistic simulation aspects, several scenarios with a team of three low-cost underwater drones (BlueROV-1) are presented and discussed. Consequently, the main contributions of this paper can be summarized as follows.

- We have proposed an incorporation of distributed adaptive neural networks control and collision-obstacle avoidance so that a group of the underwater drones is able to operate independently and autonomously in hazardous environments.
- The desired formation is proposed as a time-variant function, so that a group of UUVs can quickly change their shape between them to cross a narrow area.
- The implementation of algorithms is integrated on the Gazebo underwater drone models. The results show that the control framework can be applicable to low-cost UUVs.

The paper is organized as follows. In Section 2, preliminaries and problem formulation are introduced. Section 3 illustrates the incorporation of DANNC and collision-obstacle avoidance for a group of UUVs. Section 4 presents some experiments with the Gazebo UUV models. Finally, the conclusions are presented in Section 5.

2. Preliminaries and Problem Formulation

2.1. Preliminaries

2.1.1. Graph Theory

The interaction of a group of unmanned underwater vehicles is naturally modeled by a graph. For this reason, the vehicles can be represented as nodes (or vertices) of a graph, and the interactions, such as sensing and communication among the vehicles, can be represented as edges of the graph. Graph theory is a useful mathematical tool to design a distributed control for multi vehicle systems. A connected directed/undirected graph G can be mathematically denoted as, $G = \{\mathcal{V}, \mathcal{E}, A\}$, where $\mathcal{V}$ is the vertices set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edges set, A is the adjacency matrix of G with the elements a_{ij}, which is defined as $a_{ii} = 0$ and $a_{ij} > 0$ if the j^{th} vehicle interacts with the i^{th} vehicle, otherwise $a_{ij} = 0$. A matrix $L = [\ell_{ij}] \in \mathbb{R}^{n \times n}$ is the Laplacian of a graph G, which be defined as, $\ell_{ii} = \sum_{j=1}^{n} a_{ij}$ and $\ell_{ij} = -a_{ij}\ \forall i \neq j$.

2.1.2. Radial Basis Functions

Neural networks are generally used as approximation models for the unknown nonlinearities in the control community. In this paper, we choose the radial basis functions (RBF) neural network to compensate for the unknown items of the dynamics UUV. A RBF neural network has three layers: the input layer, the hidden layer, and the output layer. The hidden layer consists of an array of computing units called hidden nodes, which are activated by a radial basis function. Some advantages of RBF are a fast computing speed suitable for micro-controller systems, and an approximation that is good enough in the control field. A RBF network can be described by

$$f_i(\mathbf{x}_i) = W_i^{\top} \varphi_i(\mathbf{x}_i) + \varepsilon_i, \qquad \forall \mathbf{x}_i \in D \tag{1}$$

where $W_i \in R^{s \times m}$ is the weight matrix of neural network, and satisfies $\|W_i\| \leq W_{iM}$ with $W_{iM} \in R$ is a positive constant; s is the number of neurons; $\mathbf{x}_i$ is the input vector dimension m; $\varphi_i(\bullet) : R^m \rightarrow R^s$ is an activation function with the form $\varphi_i(x_i) = [\varphi_{i1}(x_i), \varphi_{i2}(x_i), ..., \varphi_{is}(x_i)]^{\top}$, $\|\varphi_i(x_i)\| \leq \varphi_{iM}$ with $\varphi_{iM} \in R$ being a positive constant; ε_i is the approximation error, $\|\varepsilon_i\| \leq \varepsilon_{iM}$ with $\varepsilon_{iM} \in R$ a positive constant; $D \subset R^n$ is a sufficiently large domain.

2.1.3. Low-Cost Underwater Drone Modeling

From the field of guidance, navigation and control of underwater vehicles, the 6 degrees of freedom (DoF) dynamic model of UUVs in body frame [26], can be written in the following form:

$$\dot{\eta}_i = J(\eta_i)\, \mathbf{v}_i \tag{2}$$

$$M_i \dot{\mathbf{v}}_{\mathbf{i}} + C_i(\mathbf{v}_i)\, \mathbf{v}_i + D_i(\mathbf{v}_i)\, \mathbf{v}_i + g_i(\eta_i) = \tau_{iAct} + \tau_{iDist} \tag{3}$$

where: $\eta_i = (x_i, y_i, z_i, \phi_i, \theta_i, \psi_i)^T$ is the position (NED: North, East, and Down) and the orientation (Euler RPY: Roll, Pitch, Yaw angles) of i^{th} UUV; $\mathbf{v}_i = [u_i, v_i, w_i, p_i, q_i, r_i]^T$ is the velocity and angular velocity; $J(\eta)$ is the *Jacobian* matrix typed 6×6 for η; $M = M_{RB} + M_A$ is a matrix, which denotes the 6×6 system inertia matrix containing M_{RB}—the generalized constant inertia matrix, and M_A—the added mass inertia matrix; $C(\mathbf{v}_i) = C_{RB}(\mathbf{v}_i) + C_A(\mathbf{v}_i)$ is the 6×6 Coriolis and centripetal forces matrix, including added mass; linear and nonlinear hydrodynamics damping are contained within the 6×6 matrix $D(\mathbf{v}_i) = D + D_n(\mathbf{v}_i)$, D contains the linear damping terms, and $D_n(\mathbf{v}_i)$ contains the nonlinear damping terms; $g(\eta)$ is the 6×1 vector of gravitational and buoyancy effects; τ_{Act} is the 6×1 control input forces and torques; and finally, τ_{Dist} is the 6×1 vector external disturbances caused by wind, waves, and ocean current.

2.2. Problem Formulation

Assumption 1. *The UUV only moves on the horizontal plane. This means that the robot dives to a predetermined depth and always remains at this depth (see Figure 1). Moreover, the UUV moves at a low-speed. This implies that the dynamics of UUV associated with the motion in heave, roll, and pitch are neglected, that is $w = p = q \simeq 0$, and $r \simeq 0$. Then, $\eta_i = (x_i, y_i, \psi_i)^T \in R^3$ and $\mathbf{v}_i = [u_i, v_i, r_i]^T \in R^3$.*

Assumption 2. *The graph G is directed and connected, and the leading UUV has access to the reference trajectory. The following UUVs always sense the leading UUV, and rely on the cameras equipped on each UUV.*

Assumption 3. *In this work, we only focus on the control for the follower UUVs with the assumption that the leader UUV is controlled in a given trajectory.*

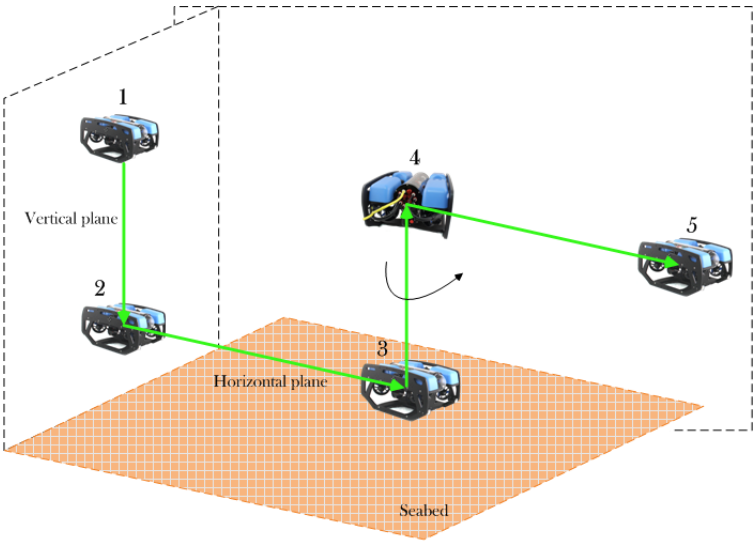


Figure 1. A proposal for submarine pipeline monitoring by using a low-cost unmanned underwater vehicle (UUV).

According to Fossen et al. [27], new coordinates, namely, the vessel parallel coordinates, are introduced as

$$\eta_{ib} = R^{\top}(\psi_i)\,\eta_i \tag{4}$$

where $\eta_{ip} \in R^3$ is the position and heading of UUV in the body-fixed coordinates. Consequently, under the assumption 1 with $r_i = \dot{\psi}_i \simeq 0$, Equations (2) and (3) can be expressed in terms of η_{ip} by

$$\dot{\eta}_{ib} = \mathbf{v}_i \tag{5}$$

$$M_i\dot{\mathbf{v}}_\mathbf{i} + C_i(\mathbf{v}_i)\,\mathbf{v}_i + D_i(\mathbf{v}_i)\,\mathbf{v}_i + g_i(\eta_i) = \tau_{iAct} + \tau_{iDist} \tag{6}$$

then

$$\dot{\eta}_{ib} = \mathbf{v}_i \tag{7}$$

$$\dot{\mathbf{v}}_\mathbf{i} = M_i^{-1}[\tau_{iAct} + \tau_{iDist} - C_i(\mathbf{v}_i)\,\mathbf{v}_i - D_i(\mathbf{v}_i)\,\mathbf{v}_i - g_i(\eta_i)] \tag{8}$$

Let $\mathbf{x}_i = (\eta_{ib}, \mathbf{v}_i)^{\top} = (x_{ib}, y_{ib}, \psi_{ib}, u_i, v_i, r_i)^{\top} \in R^6$ is the state of UUV i in the body-fixed coordinates, Equations (7) and (8) can be written in a state-space model with a form

$$\dot{\mathbf{x}}_i = A\mathbf{x}_i + B\left[u_i + f_i(\mathbf{x}_i) + w_i(t)\right] \tag{9}$$

where

$$A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, \qquad B = \begin{bmatrix} 0_{3\times3} \\ M_i^{-1} \end{bmatrix} \tag{10}$$

$w_i = \tau_{iDist}$, $u_i = \tau_{iAct} \in R^3$, and $f_i(\mathbf{x}_i) = -C_i(\mathbf{v}_i)\mathbf{v}_i - D_i(\mathbf{v}_i)\mathbf{v}_i - g_i(\mathbf{v}_i)$ is the unmodeled and unknown terms of the dynamics UUV i, which is approximated by Equation (1).

Let $\delta_{ij} = [\delta_{x,ij}, \delta_{y,ij}, \delta_{\psi,ij}, 0, 0, 0] \in R^6$ denote the desired state deviation between UUV i and UUV j in 3 DOF. In addition, the reference model is considered as the leading UUV, which is given by

$$\dot{\mathbf{x}}_L = A\mathbf{x}_L + Br(\mathbf{x}_L, t) \tag{11}$$

The objective of this paper is to apply a distributed adaptive neural network for the group of low-cost underwater drones to achieve the desired formation δ_{ij} and to stabilize the attitude dynamics $d_{ij} \to \delta_{ij}$. Meanwhile, the distance between the UUV d_{ij} is guaranteed to avoid collision $d_{in} < d_{ij}$ with

$d_{in} < D_{out} < \delta_{ij}$, where d_{in} and D_{out} are predefined values. And finally, $\max(d_{ij}) < (h - \Delta)$ to ensure a group of UUVs can pass the narrow area, where h is the smallest size between narrow slots, Δ is a given value depending on the size of UUV.

Remark 1. *To satisfy the assumption that the graph is connected in practice with the use of cameras, which are equipped on each UUV, a proposal for the following UUV sense of relative position and orientation for the leading UUV is introduced in this paper.*

3. Incorporation of DANNC and Collision-Obstacle Avoidance for a Group of UUVs

In this section, we first introduce the overall distributed adaptive neural networks and apply to the formation tracking problem in Section 3.1. Then we integrate the formation tracking problem with collision and obstacle avoidance in Sections 3.2 and 3.3.

3.1. Leader-Follower Formation Tracking

For the following UUV i, the control input consists of four terms, which are the formation tracking control input u_{iFC}, the robust control input u_{iR}, the neural network control input u_{iNN}, and the collision avoidance u_{iCA} (see in Figure 2).

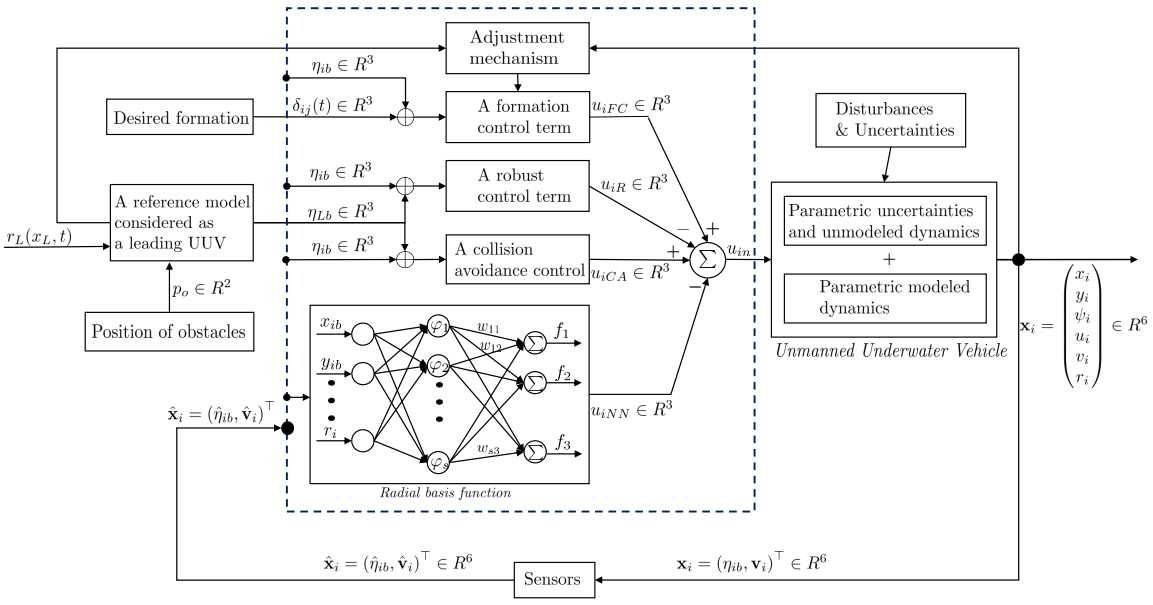


Figure 2. The overall distributed adaptive neural networks formation tracking control with collision and obstacle avoidance for the UUV follower.

3.1.1. Formation Control Term

To do a formation control, we define an auxiliary variable vector, $e_i \in R^6$ for the UUV i as

$$e_i = \sum_{j \in N_i} a_{ij} \left(\mathbf{x}_i - \delta_i - \left(\mathbf{x}_j - \delta_j \right) \right) + a_{iL} \left(\mathbf{x}_i - \delta_i - \mathbf{x}_L \right) \tag{12}$$

where a_{ij} and a_{iL} are described in Section 2.1.1.

A distributed adaptive formation tracking control is then designed based on the consensus tracking protocol and the feedback linearization technique, which is given as

$$u_{iC} = c_i K e_i \tag{13}$$

where $K = -B^\top P \in R^{3\times 6}$ is a feedback control gain, $c_i \in R$ is a coupling weight updated as

$$\dot{c}_i = \Gamma_{ic} \Pr\left(c_i, e_i^\top PBB^\top Pe_i\right) \tag{14}$$

where Γ_{ic} is a positive constant, and P is the positive definite solution based on the algebraic Riccati equation

$$A^\top P + PA + Q - PBB^\top P = 0 \tag{15}$$

where $Q \in R^{6\times 6}$ is a positive-definite matrix.

3.1.2. Neural Network Control Term

Because the dynamics of UUV have unmodeled and unknown terms, a neural network control u_{iNN} is introduced to compensate for these terms. Each following UUV will have a neural network term locally to keep track of the current estimates. Therefore, the control input for UUV i in Equation (13) is the added u_{iNN} term, which is given by following

$$u_{in} = u_{iC} - u_{iNN} \tag{16}$$

with

$$u_{iNN} = \hat{W}_i^\top \varphi\left(\mathbf{x}_i\right) \tag{17}$$

where $\hat{W}_i \in R^{s_i\times m}$ is the estimation of W_i, s_i is the number of neural network neurons for the UUV i, m is the number of input values, and the input vector $\mathbf{x}_i = \left(\eta_{ib}, \mathbf{v}_i\right)^\top = \left(x_{ib}, y_{ib}, \psi_{ib}, u_i, v_i, r_i\right)^\top \in R^6$. That means $m = 6$ in this work. The activation function $\varphi(\mathbf{x}_i) \in R^{s\times 1}$.

To guarantee the weight matrices of neural network retain bounded, the projection algorithm [5,28] is adopted to update the parameters

$$\dot{\hat{W}}_i = \Gamma_{iW} \Pr\left(\hat{W}_i, \varphi_i\left(\mathbf{x}_i\right) e_i^\top PB\right) \tag{18}$$

where $\Gamma_{iW} \in R$ is a positive constant. Pr is the projection operator (a matrix $n \times m$), with the general form described as

$$\Pr(\boldsymbol{\Theta}, \mathbf{Y}, \boldsymbol{\Phi}) = [\Pr(\theta_1, y_1, \phi_1), ..., \Pr(\theta_m, y_m, \phi_m)] \tag{19}$$

with

$$\Pr(\theta_i, y_i, \phi_i) = \begin{cases} y_i - \dfrac{\bigtriangledown\phi_i(\theta_i)(\bigtriangledown\phi_i(\theta_i))^\top}{\| \bigtriangledown\phi_i(\theta_i) \|^2} y_i\phi_i(\theta_i) & \text{if } \phi_i(\theta_i) > 0 \wedge y_i^\top \bigtriangledown \phi_i(\theta_i) > 0 \\ y_i & \text{otherwise} \end{cases} \tag{20}$$

where $i = 1$ to m; $\Theta = \begin{bmatrix} \theta_1 & \cdots & \theta_m \end{bmatrix} \in R^{s\times m}$; $\mathbf{Y} = \begin{bmatrix} y_1 & \cdots & y_m \end{bmatrix} \in R^{s\times m}$; $\boldsymbol{\Phi} = \begin{bmatrix} \phi_1 & \cdots & \phi_m \end{bmatrix} \in R^{s\times m}$; $\phi_i : R^k \to R$ is a convex function.

In this work, ϕ_i is chosen as

$$\phi_i(\theta_i) = \frac{\theta_i^\top\theta - \theta_{iM}^2}{\epsilon_{\theta_i}\theta_{iM}^2} \tag{21}$$

with $\theta_{iM} \in R$ is a projection norm bound, and ϵ_{θ_i} is a projection tolerance bound.

Lemma 1 ([28]). *The projection operator has an important property, which is given as*

$$trace\{(\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta})^\top (\Pr(\hat{\boldsymbol{\Theta}}, \mathbf{Y}, \boldsymbol{\Phi}) - \mathbf{Y})\} \leq 0 \tag{22}$$

The objective here is to use the states from the neighbors of UUV i to evaluate the performance of the current control of UUV i with the current estimates $\hat{f}_i(\mathbf{x}_i)$ of the function $f_i(\mathbf{x}_i)$. This is done through online adjustment of the weight of the neural network $\hat{W}_i$ in Equation (18).

3.1.3. Robust Control Term

According to [7], a robust term is designed to guarantee the tracking error achieve a asymptotic convergence. It is given as

$$u_{iR} = \frac{S_i^2 B^\top P e_i}{S_i \left\| B^\top P e_i \right\| + e^{-t}} \tag{23}$$

with S_i is updated by :

$$S_i = \Gamma_{iS} \int_0^t \left\| e_i^\top P B \right\| dt \tag{24}$$

where $\Gamma_{iS} \in \Re$ is a positive constant.

Remark 2. *For the proof of a stability, the readers are referred to [7].*

Combining Equations (16) and (23), we have a formation tracking control, which is given as

$$u_{in} = u_{iC} - u_{iNN} - u_{iR} \tag{25}$$

Remark 3. *The robust control u_{iR} is designed by using only the local information, which are the relative states between UUVs. This is essential for applications of low-cost underwater drones, where positioning and communication are limited.*

3.2. Collision Avoidance for a Group of Multiple UUVs

By defining two values d_{in} and D_{out}, as shown in Figure 3, the collision avoidance between UUVs will be guaranteed, and the change of distance will not happen suddenly. Specifically, when these distances are between d_{in} and D_{out}, a control function will be activated.

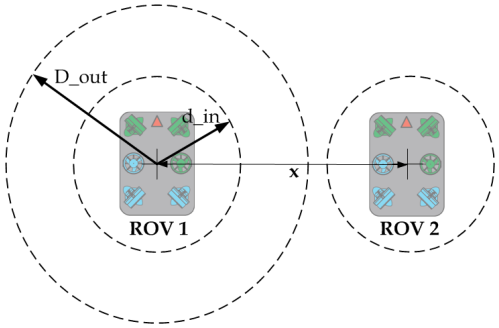


Figure 3. Defines the zone to avoid collisions between UUVs.

Let $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ be the distance between UUV i and j. According to [18], a novel adaptive repulsive potential with finite cut-off D_{out} is given as

$$\psi_c \left(d_{ij} \right) = \int_{D_{out}}^{d_{ij}} \phi \left(s \right) ds$$

with

$$\phi(d_{ij}) = \begin{cases} \dfrac{-d_{ij}}{\sigma_{ij} + {d_{ij}}^2} & \text{if } d_{ij} \in (0, d_{in}) \\ \dfrac{-d_{ij}}{2\left(\sigma_{ij} + {d_{ij}}^2\right)} \left[1 + \cos\left(\pi \dfrac{d_{ij} - d_{in}}{D_{out} - d_{in}} \right) \right] & \text{if } d_{ij} \in [d_{in}, D_{out}] \\ 0 & \text{if } d_{ij} \notin (0, D_{out}] \end{cases}$$

where the parameter σ_{ij} can be tuned online by each UUV at the time in the zone of collision avoidance. Then, a smooth collective potential function is defined as

$$V\left(d_{ij}\right) = \frac{1}{2}\sum_{i}\sum_{j\in N_i}\psi_c\left(d_{ij}\right) \tag{26}$$

Finally, the collision avoidance term u_{iCA} can be written as

$$u_{iCA} = -\nabla_{d_{ij}}V\left(d_{ij}\right) = \sum_{j\in N_i}\phi\left(d_{ij}\right)\frac{\mathbf{p}_j - \mathbf{p}_i}{d_{ij}} \tag{27}$$

where $\mathbf{p}_i = (x_i, y_i)^\top$ and $\mathbf{p}_j = (x_j, y_j)^\top$ are the positions of UUV i and j, respectively.

From Equations (25) and (27), we have a formation tracking control with a collision avoidance term, which is given as

$$u_{in} = u_{iC} - u_{iNN} - u_{iR} + u_{iCA} \tag{28}$$

3.3. Obstacle Avoidance for a Group of Multiple UUVs

For operations in hazardous environments, we define two environment zones (see Figure 4). Zone 1 is an environment with the obstacles, where the group of UUVs will change a trajectory to avoid obstacles. Zone 2 is an environment with a narrow area, where the group of UUVs will change the desired size of formation to be able to overcome that area.

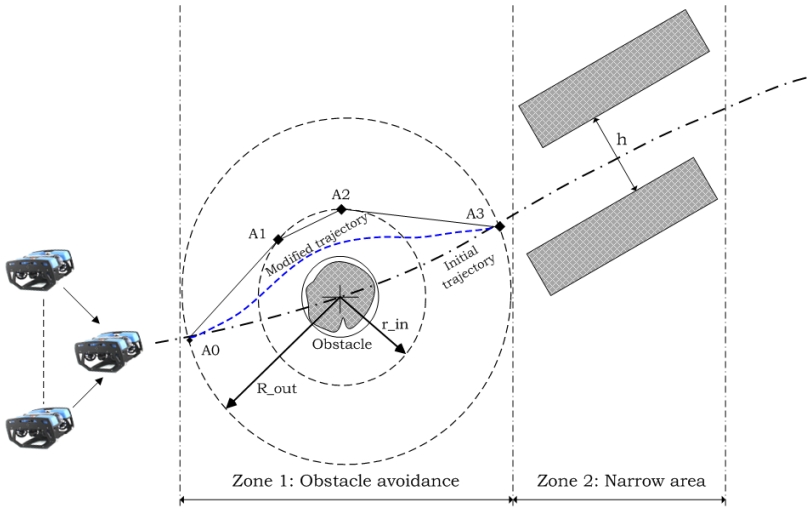


Figure 4. Partition of two regions of a UUV for obstacle avoidance.

Assumption 4. *For zone 1, the static obstacles or dynamic obstacles are identified by the vision system or the sonar sensors. That means the distance from the obstacles to the UUV is predefined and updated continuously.*

A smooth curve can be defined using the control point $A0, A1, A2, A3$, as shown in Figure 4. Note that the Bézier curve does not pass through the intermediate points $A2, A3$, however, it passes through the points $A0$ and $A3$. The implementation steps are outlined as Algorithm 1. For more details, the readers can refer to the previous works [29].

Assumption 5. *For zone 2, the position and distance h of narrow area are predefined through sensors.*

By defining a desired formation as time-variant function, we have

$$\delta_{ij} = \delta_{ij}\left(t\right) = \delta_j(t) - \delta_i(t) \tag{29}$$

Algorithm 1: Propose to make a modified trajectory for obstacle avoidance.

```
Data: Distance from UUV to obstacle r
Result: Trajectory for UUV
while r <= R_out do
    read current;
    Determine points A0, A1, A2, A3;
    if r <= r_in then
        Make a modified trajectory using the Bézier curve ;
        UUV follow the modified trajectory;
    else
        go back to the initial trajectory;
    end
end
```

The idea here is to compare the distance from the UUV leader to the narrow area. When the UUV leader is in a narrow area, a group of UUVs will compare a desired formation $\delta_{ij}(t)$ with the distance from the narrow area h. It will have to satisfy the following conditions

$$\delta_{ij}(t) = \begin{cases} \delta_{ij}(t) & \text{if } \delta_{ij}(t) < h - \Delta \\ h - \Delta & \text{if } \delta_{ij}(t) \geq h - \Delta \end{cases} \tag{30}$$

where Δ is a given value, which depends on the size of the UUV to ensure that the group of UUV moves through a narrow area without collision, and h is the smallest distance in the narrow area.

4. Experiments with a Group of Low-Cost UUVs

4.1. Experimental Setup

The direct graph of three UUVs is shown in Figure 4. The distributed adaptive neural network control and the collision and obstacle avoidance have been tested and implemented in UUV-Simulator [23], which is based on ROS/Gazebo. The objective of the proposed experiments is to coordinate the low-cost underwater drones by simultaneously controlling some of their local information, and introducing a complete operating scenario in hazardous environments. We consider a leader–follower scheme composed of three identical underwater drones. The model that was used for the simulation is a BlueROV-1, actuated in surge, sway, heave, and yaw via a 6 thruster set configuration (the motion in roll and pitch has been neglected as both DoFs are passive and minimally affect the others). According to [30], the main inertial and hydrodynamic parameters of BlueROV-1 are given by

$$M = M_{RB} + M_A = \begin{bmatrix} 12.81 & 0 & 0 \\ 0 & 20.01 & 0 \\ 0 & 0 & 0.12 \end{bmatrix}, \quad D = \begin{bmatrix} -28.27 & 0 & 0 \\ 0 & -134.74 & 0 \\ 0 & 0 & -0.07 \end{bmatrix} \tag{31}$$

then,

$$B = \begin{bmatrix} 0_{3\times 3} \\ M_i^{-1} \end{bmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.0781 & 0 & 0 \\ 0 & 0.0050 & 0 \\ 0 & 0 & 8.3333 \end{pmatrix} \tag{32}$$

and the mass of BlueROV-1 is $m = 7.31$ kg, the length is 483 mm, the width is 330 mm, and the height is 267 mm. The unmodeled dynamics f_i and disturbances w_i are considered randomly as

$$f_i(x_i) = \begin{bmatrix} -0.01u_i v_i^2 \\ -0.02u_i v_i \\ -0.01r_i u_i \end{bmatrix} \text{ and } w_i = \begin{bmatrix} -0.1\sin(0.5t)\cos(0.3t) \\ -0.1\sin(0.2t)\cos(0.5t) \\ -0.1\sin(0.1t)\cos(0.6t) \end{bmatrix} \tag{33}$$

The initial states of three BlueROV-1 are given by $\mathbf{x}_1 = [6, 0, \pi/2, 0, 0, 0]^\top$, $\mathbf{x}_2 = [-6, 1, 0, 0, 0, 0]^\top$, $\mathbf{x}_3 = [0, 3, \pi/2, 0, 0, 0]^\top$. The desired formation is given by $\delta_1 = [5\sqrt{3}, 0, 0, 0, 0, 0]^\top$, $\delta_2 = [0, 5, 0, 0, 0, 0]^\top$, $\delta_3 = [0, -5, 0, 0, 0, 0]^\top$. The distance of a narrow area is chosen as $h = 8$ m, $\Delta = 2$ m.

The number of neural network neurons for each UUV has adjusted to 10 neurons. The number of input values for neural network is 6. The neural network activation function is chosen as a log-sigmoid of the form $\phi(\mathbf{x}_i) = 1/(1 + e^{-\kappa t})$, where κ is a positive constant ($\kappa = 2$ in this work).

From Equations (10), (15), and (32), a solution P of the Riccati equation with $Q = \text{diag}\{1, 1, 1, 1, 1, 1\}$ is

$$P = \begin{pmatrix} 5.1595 & 0 & 0 & 12.8100 & 0 & 0 \\ 0 & 6.4047 & 0 & 0 & 20.0100 & 0 \\ 0 & 0 & 1.1136 & 0 & 0 & 0.1200 \\ 12.8100 & 0 & 0 & 66.0926 & 0 & 0 \\ 0 & 20.0100 & 0 & 0 & 128.1578 & 0 \\ 0 & 0 & 0.1200 & 0 & 0 & 0.1336 \end{pmatrix} \tag{34}$$

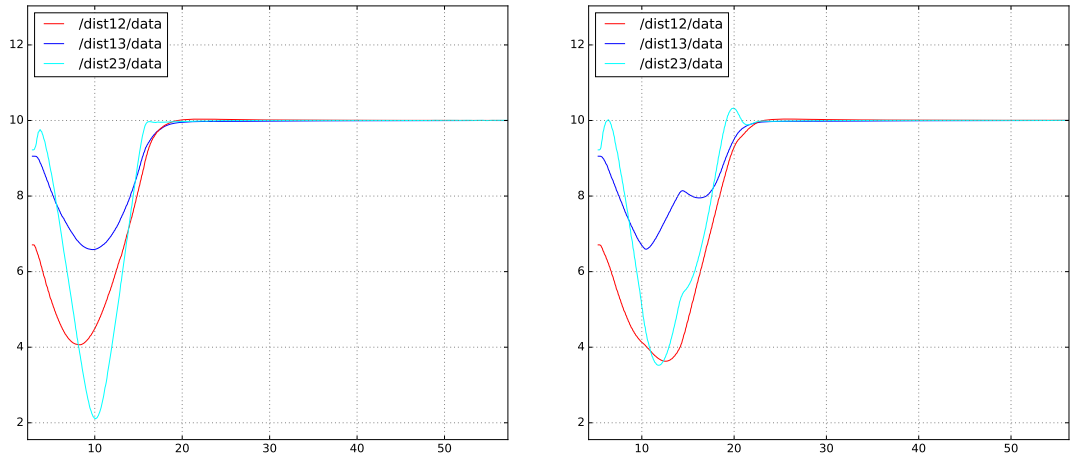
where P is a positive-definite matrix, then,

$$K = -B^\top P = \begin{pmatrix} -1.0000 & 0 & 0 & -5.1595 & 0 & 0 \\ 0 & -1.0000 & 0 & 0 & -6.4047 & 0 \\ 0 & 0 & -1.0000 & 0 & 0 & -1.1136 \end{pmatrix} \tag{35}$$

The leader UUV assumes that it has been controlled to follow the trajectory with the way-points $(0, 6)$, $(0, 12)$, $(30, 12)$, $(30, 6)$. The position of the obstacle is $(15, 9)$ and $D_{out} = 7$ m, $d_{in} = 4$ m.

4.2. Results

Figure 5 shows the distances among three BlueROV-1 drones in case of using collision avoidance terms compared to those in none of the collision avoidance term.

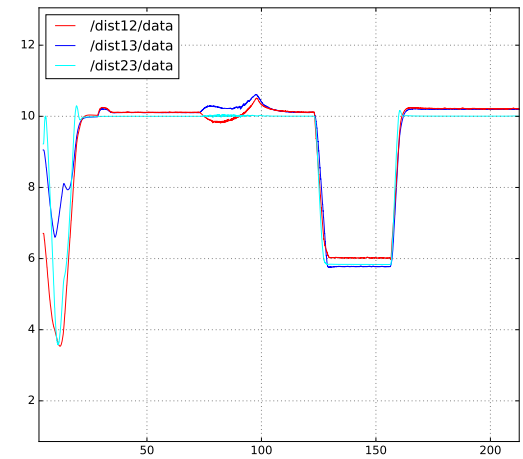


(**a**) Distances without a collision avoidance term (**b**) Distances with a collision avoidance term

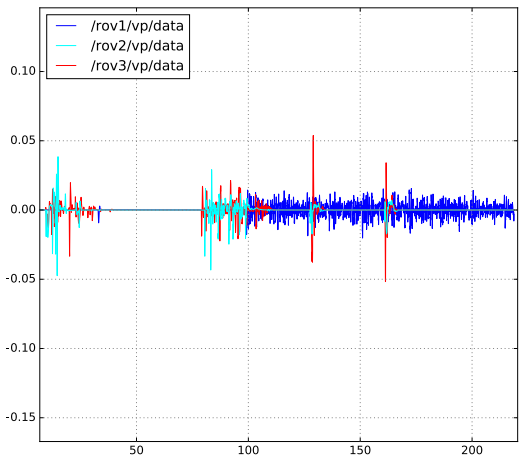
Figure 5. Distance among three BlueROV-1 in two experimental cases.

We first examine the formation control without a collision avoidance term. It can be observed that the minimum distance among the UUV 2 and the UUV 3 is $d_{23} = 2$ m (see Figure 5a). That means there is a collision between the two UUVs. We then examine the formation control when applying the collision avoidance term. Figure 5b shows that the minimum distance between robots is and $d_{23} \approx 4$ m. This guarantees no collision when the parameters selected are $D_{out} = 7$ m, $d_{in} = 4$ m, and $\sigma_{ij} = 3$.

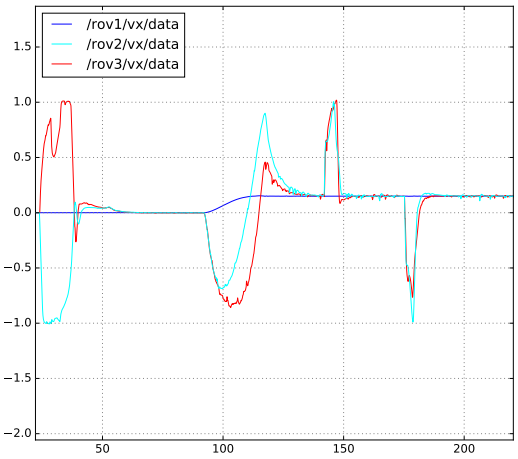
We further examine the distributed formation tracking control with collision in hazard environments. A group of UUVs start from a random configuration, reach a desired formation, and then follow a trajectory of the UUV-leader. During a move, a group of UUVs can change the shape of formation to cross a narrow area. Figure 6 shows the complete simulation scenario with three BlueROV-1. The distances among the UUVs are shown in Figure 6a, which converges to 10 m. It is clearly shown that the UUV follower is indeed tracking to the leader's position, and maintaining the desired distances with the same heading angle. The velocities of three BlueROV-1 are illustrated in Figure 6b–d, respectively. In addition, during the period from $t = 120$ s $\rightarrow$ $t = 170$ s, it can be seen that the group of UUV changed the distance $d_{ij} = 6$ m to cross narrow areas. It should be noted that the velocity of the UUV is not smooth along the axes, but this result is still acceptable. The readers can find more details in Appendix A (see Figures A1 and A2).



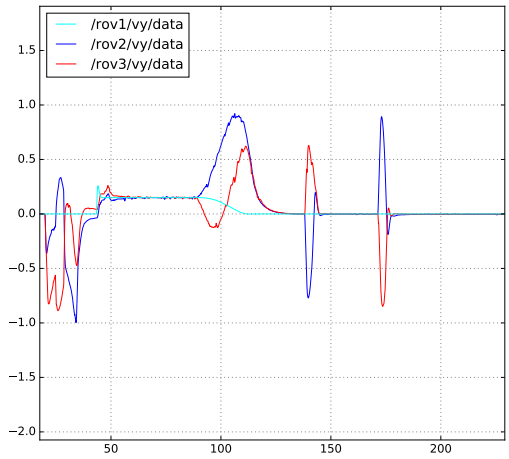
(**a**) Distances among three BlueROV-1



(**b**) Angular velocity $r_i \simeq 0$



(**c**) Velocities in the axis x

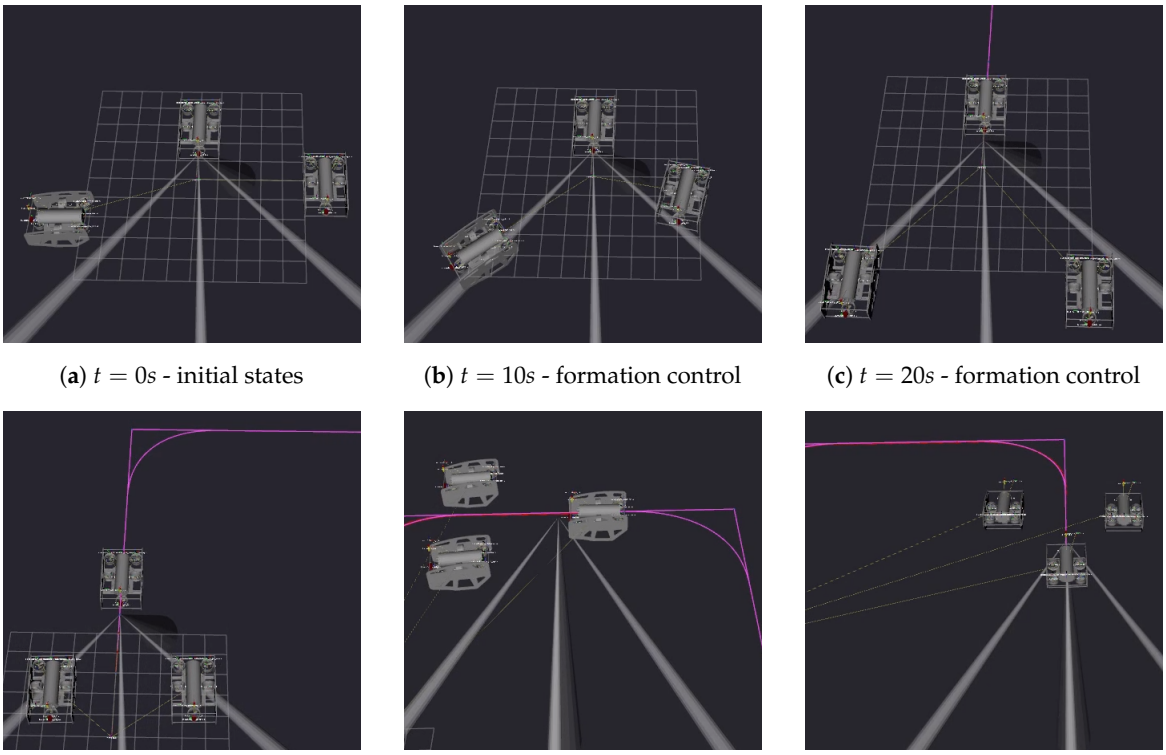


(**d**) Velocities in the axis y

Figure 6. A complete simulation scenario with three BlueROV-1.

An example of the evolution of three BlueROV-1 in ROS/Gazebo is presented in Figure 7, where a visualization of BlueROV-1 dynamics is presented under the underwater hydrostatic and

hydrodynamic effects. Finally, we have also performed some experiments with a modified trajectory based on Algorithm 1 to avoid obstacles (see Figure A3 in Appendix A).



(**a**) $t = 0s$ - initial states (**b**) $t = 10s$ - formation control (**c**) $t = 20s$ - formation control

(**d**) $t = 30s$ - change of a formation (**e**) $t = 125s$ - trajectory tracking (**f**) $t = 200s$

Figure 7. An example of the evolution of three BlueROV-1 in robot operating system (ROS)/Gazebo.

5. Conclusions

The formation tracking control problems of a group of low-cost underwater drones are subject to nonlinearity, parametric uncertainties, and external disturbances have been discussed and implemented. A complete scenario of the underwater drone operation has been simulated by integrating the formation tracking control term and the collision-obstacle avoidance term. In particular, the use of a neural network technique approximates unknown parameters of UUV, thereby designing controllers to compensate for these parameters. We emphasize here that this paper aims to take the first step for future research to develop an application of multiple low-cost underwater drones.

Several important issues along this research are to be addressed in the future, such as performing experiments in a pool with image processing algorithms to determine the relative position of the UUV follower and UUV leader, and the integration of some low-cost sensors (i.e., a sonar, an underwater LiDAR) to increase accuracy in determining the distance between UUVs.

Although the approach is distributed, due to its implementation on the ROV-1 platform, the calculations are still centralized in the implementation aspect. However, with ROV-2 being released, the computations will be distributed completely for each UUV. In addition, we have only studied a group of UUVs on three degrees of freedom (3-DOF). In the future, the research will be extended to 6-DOF, where the effect of depth control of UUVs will be investigated.

Author Contributions: Conceptualization, H.A.P., S.T. and V.H.N.; methodology, H.A.P., T.S. and V.H.N.; software, H.A.P., T.S.; validation, H.A.P., T.S., V.H.N. and V.G.; writing—original draft preparation, H.A.P. and T.S.; writing—review and editing, H.A.P., T.S. and V.H.N.; visualization, H.A.P. and T.S.; supervision, T.S.; project administration, T.S.; funding acquisition, T.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ministry of National Education France.

Acknowledgments: We would like to express our sincere appreciation for the insightful and valuable comments of the editors and reviewers. Those allow us to clarify and improve the presentation of this work.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

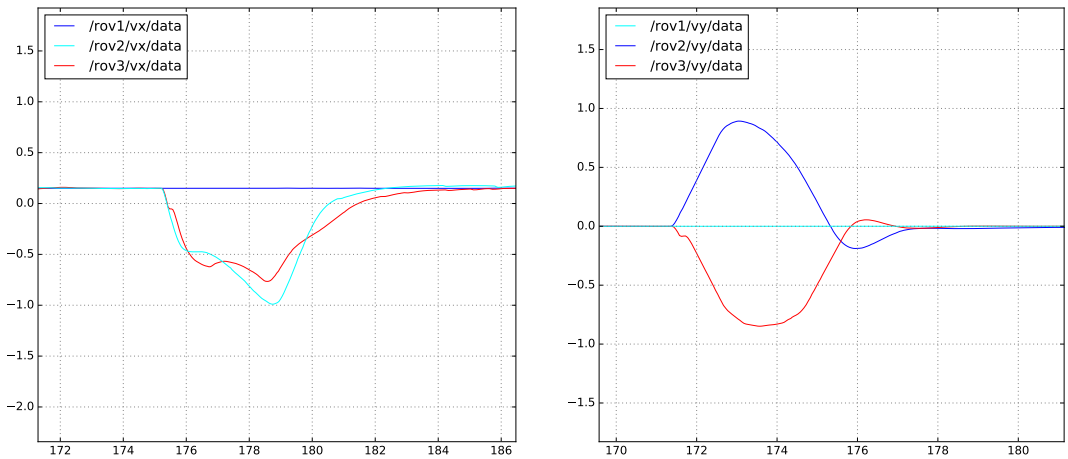
The following abbreviations are used in this manuscript:

AUV	Autonomous Underwater Vehicle
DANNC	Distributed Adaptive Neural Network Control
LiDAR	Light Detection and Ranging
NN	Neural Network
ROS	Robot Operating System
ROV	Remotely Operated Underwater Vehicle
UUV	Unmanned Underwater Vehicle
UWSim	UnderWater Simulator

Appendix A

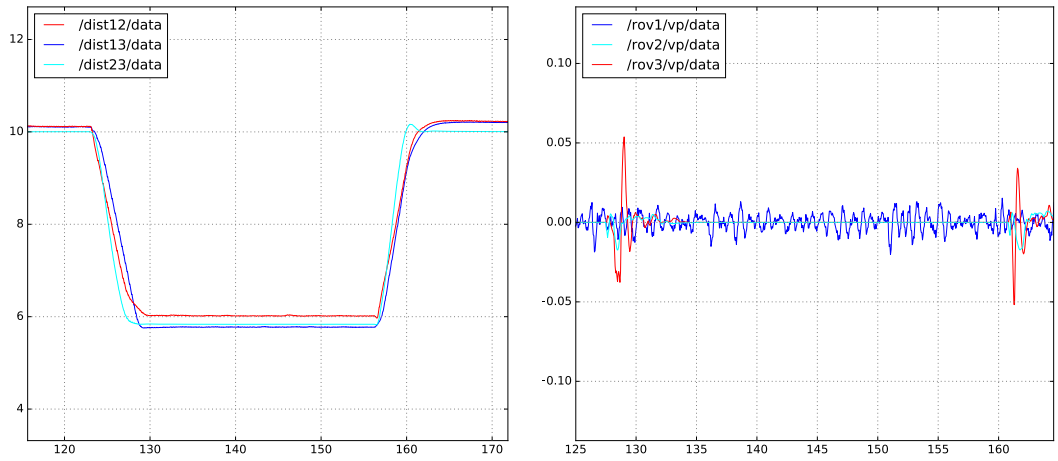
In this section we would like to add the velocity and distance of the UUV while crossing the narrow area. It should be noted that the timing is asynchronous in Figures A1a,b and A2a,b because the starting point for the data collection process is not synchronized in ROS/Gazebo. However, this does not affect simulation results.

Figure A3 shows the distances from the UUV i to the obstacle with an initial trajectory and a modified trajectory. It can be seen that the distance from the UUV 3 to the obstacle is 0.5 m, this is defined as the collision between UUV and obstacle (see Figure A3a). By using a modified trajectory, the distance from the UUV i to the obstacle is $>$ 2.8 m, this proves no collision (see Figure A3b).



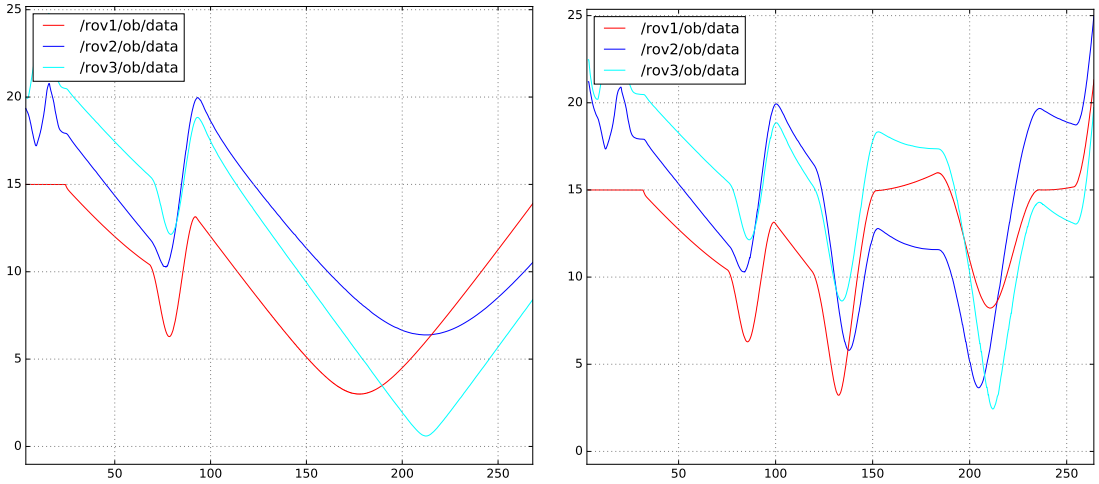
(**a**) Velocities in the axis x (**b**) Velocities in the axis y

Figure A1. Velocities of three BlueROV-1 while crossing a narrow area.



(**a**) Distances between three BlueROV-1 (**b**) Angular velocity of three BlueROV-1

Figure A2. Velocities and distances of three BlueROV-1 while crossing a narrow area.



(**a**) With an initial trajectory (**b**) With a modified trajectory

Figure A3. The distances from the UUV *i* to the obstacle.

References

1. Liu, H.; Wang, Y.; Lewis, F.L. Robust Distributed Formation Controller Design for a Group of Unmanned Underwater Vehicles. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, 1–9. doi:10.1109/TSMC.2019.2895499. [CrossRef]
2. Bechlioulis, C.P.; Giagkas, F.; Karras, G.C.; Kyriakopoulos, K.J. Robust Formation Control for Multiple Underwater Vehicles. *Front. Robot. AI* **2019**, *6*, doi:10.3389/frobt.2019.00090. [CrossRef]
3. Yuan, C.; Licht, S.; He, H. Formation Learning Control of Multiple Autonomous Underwater Vehicles With Heterogeneous Nonlinear Uncertain Dynamics. *IEEE Trans. Cybern.* **2018**, *48*, 2920–2934. doi:10.1109/tcyb.2017.2752458. [CrossRef] [PubMed]
4. Elhaki, O.; Shojaei, K. Neural network-based target tracking control of underactuated autonomous underwater vehicles with a prescribed performance. *Ocean Eng.* **2018**, *167*, 239–256. doi:10.1016/j.oceaneng.2018.08.007. [CrossRef]
5. Liu, Y.; Huang, P.; Zhang, F.; Zhao, Y. Distributed Formation Control Using Artificial Potentials and Neural Network for Constrained Multiagent Systems. *IEEE Trans. Control Syst. Technol.* **2018**, 1–8. doi:10.1109/tcst.2018.2884226. [CrossRef]

6. Cui, R.; Yang, C.; Li, Y.; Sharma, S. Adaptive Neural Network Control of AUVs With Control Input Nonlinearities Using Reinforcement Learning. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *47*, 1019–1029. doi:10.1109/tsmc.2016.2645699. [CrossRef]
7. Peng, Z.; Wang, H.; Wang, D.; Sun, G.; Zhang, H. Distributed model reference adaptive control for cooperative tracking of uncertain dynamical multi-agent systems. *IET Control Theory Appl.* **2013**, *7*, 1079–1087. doi:10.1049/iet-cta.2012.0765. [CrossRef]
8. Yoo, S.J. Distributed Consensus Tracking for Multiple Uncertain Nonlinear Strict-Feedback Systems Under a Directed Graph. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 666–672. doi:10.1109/tnnls.2013.2238554. [CrossRef]
9. Hou, Z.G.; Cheng, L.; Tan, M. Decentralized Robust Adaptive Control for the Multiagent System Consensus Problem Using Neural Networks. *IEEE Trans. Syst. Man. Cybern. Part B (Cybern.)* **2009**, *39*, 636–647. doi:10.1109/tsmcb.2008.2007810. [CrossRef]
10. Abreu, P.C.; Pascoal, A.M. Formation Control in the scope of the MORPH project. Part I: Theoretical Foundations. *IFAC-PapersOnLine* **2015**, *48*, 244–249. doi:10.1016/j.ifacol.2015.06.040. [CrossRef]
11. Shi, H.; Hou, M.; Wu, Y. Distributed Control for Leader-Following Consensus Problem of Second-Order Multi-Agent Systems and Its Application to Motion Synchronization. *Appl. Sci.* **2019**, *9*, 4208. doi:10.3390/app9204208. [CrossRef]
12. Jia, Z.; Wang, L.; Yu, J.; Ai, X. Distributed adaptive neural networks leader-following formation control for quadrotors with directed switching topologies. *ISA Trans.* **2019**, doi:10.1016/j.isatra.2019.02.030. [CrossRef] [PubMed]
13. Wang, D.; Zong, Q.; Tian, B.; Shao, S.; Zhang, X.; Zhao, X. Neural network disturbance observer-based distributed finite-time formation tracking control for multiple unmanned helicopters. *ISA Trans.* **2018**, *73*, 208–226. doi:10.1016/j.isatra.2017.12.011. [CrossRef] [PubMed]
14. Peng, Z.; Wen, G.; Yang, S.; Rahmani, A. Distributed consensus-based formation control for nonholonomic wheeled mobile robots using adaptive neural network. *Nonlinear Dyn.* **2016**, *86*, 605–622. doi:10.1007/s11071-016-2910-2. [CrossRef]
15. Li, Y.; Wang, C.; Cai, X.; Li, L.; Wang, G. Neural-network-based distributed adaptive asymptotically consensus tracking control for nonlinear multiagent systems with input quantization and actuator faults. *Neurocomputing* **2019**, *349*, 64–76. doi:10.1016/j.neucom.2019.04.018. [CrossRef]
16. Yang, Q.; Cao, M.; de Marina, H.G.; Fang, H.; Chen, J. Distributed formation tracking using local coordinate systems. *Syst. Control Lett.* **2018**, *111*, 70–78. doi:10.1016/j.sysconle.2017.11.004. [CrossRef]
17. Yuan, C.; Zeng, W.; Dai, S.L. Distributed model reference adaptive containment control of heterogeneous uncertain multi-agent systems. *ISA Trans.* **2019**, *86*, 73–86. doi:10.1016/j.isatra.2018.11.003. [CrossRef]
18. Mondal, A.; Behera, L.; Sahoo, S.R.; Shukla, A. A novel multi-agent formation control law with collision avoidance. *IEEE/CAA J. Autom. Sin.* **2017**, *4*, 558–568. doi:10.1109/jas.2017.7510565. [CrossRef]
19. Xia, Y.; Na, X.; Sun, Z.; Chen, J. Formation control and collision avoidance for multi-agent systems based on position estimation. *ISA Trans.* **2016**, *61*, 287–296. doi:10.1016/j.isatra.2015.12.010. [CrossRef]
20. Yu, J.; Ji, J.; Miao, Z.; Zhou, J. Formation control with collision avoidance for uncertain networked Lagrangian systems via adaptive gain techniques. *IET Control Theory Appl.* **2018**, doi:10.1049/iet-cta.2017.1065. [CrossRef]
21. Soriano-Asensi, D.C.A.; Avilés, J.V.M.; Prades, R.M.; Valero, P.J.S. Underwater Wireless Communications for Cooperative Robotics with UWSim-NET. *Appl. Sci.* **2019**, *9*, 3526.
22. Prats, M.; Pérez, J.; Fernández, J.J.; Sanz, P.J. An open source tool for simulation and supervision of underwater intervention missions. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012; pp. 2577–2582. doi:10.1109/IROS.2012.6385788. [CrossRef]
23. Manhães, M.M.M.; Scherer, S.A.; Voss, M.; Douat, L.R.; Rauschenbach, T. UUV Simulator: A Gazebo-based package for underwater intervention and multi-robot simulation. In Proceedings of the OCEANS 2016 MTS/IEEE Monterey, Monterey, CA, USA, 19–23 September 2016; pp. 1–8. doi:10.1109/OCEANS.2016.7761080. [CrossRef]
24. Martínez, N.L.; Martínez-Ortega, J.F.; Castillejo, P.; Martínez, V.B. Survey of Mission Planning and Management Architectures for Underwater Cooperative Robotics Operations. *Appl. Sci.* **2020**, *10*, 1086. doi:10.3390/app10031086. [CrossRef]

25. Martínez, N.L.; Martínez-Ortega, J.F.; Rodríguez-Molina, J.; Zhai, Z. Proposal of an Automated Mission Manager for Cooperative Autonomous Underwater Vehicles. *Appl. Sci.* **2020**, *10*, 855. doi:10.3390/app10030855. [CrossRef]
26. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley: Hoboken, NJ, USA, 2011.
27. Fossen, T.I.; Perez, T. Kalman Filtering for Positioning and Heading Control of Ships and Offshore Rigs. In *IEEE Control Systems Magazine*; IEEE: Piscataway, NJ, USA, 2009.
28. Lavretsky, E.; Gibson, T.E.; Annaswamy, A.M. Projection Operator in Adaptive Systems. *arXiv* **2012**, arXiv:1112.4232.
29. Pham, H.A.; Soriano, T.; Ngo, V.H. Integrated scenarios of formation tracking and collision avoidance of multi-vehicles. In Proceedings of the 2018 13th Annual Conference on System of Systems Engineering (SoSE), Paris, France, 19–22 June 2018; pp. 313–318. doi:10.1109/SYSOSE.2018.8428730. [CrossRef]
30. Skaalvik, S.S. System Identification and State Estimation for ROV uDrone. Master's Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2016.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).