sensors

MDPI

Article

A Novel Paradigm for Underwater Monitoring Using Mobile Sensor Networks

Anja Babić [1,*], Ivan Lončar [1], Barbara Arbanas [2], Goran Vasiljević [2], Tamara Petrović [2], Stjepan Bogdan [2] and Nikola Mišković [1]

[1] LABUST—Laboratory for Underwater Systems and Technologies, Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb, Croatia; ivan.loncar@fer.hr (I.L.); nikola.miskovic@fer.hr (N.M.)

[2] LARICS—Laboratory for Robotics and Intelligent Control Systems, Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb, Croatia; barbara.arbanas@fer.hr (B.A.); goran.vasiljevic@fer.hr (G.V.); tamara.petrovic@fer.hr (T.P.); stjepan.bogdan@fer.hr (S.B.)

* Correspondence: anja.babic@fer.hr

Received: 6 July 2020; Accepted: 11 August 2020; Published: 17 August 2020

Abstract: This paper presents a novel autonomous environmental monitoring methodology based on collaboration and collective decision-making among robotic agents in a heterogeneous swarm developed within the project subCULTron, tested in a realistic marine environment. The swarm serves as an underwater mobile sensor network for exploration and monitoring of large areas. Different robotic units enable outlier and fault detection, verification of measurements and recognition of environmental anomalies, and relocation of the swarm throughout the environment. The motion capabilities of the robots and the reconfigurability of the swarm are exploited to collect data and verify suspected anomalies, or detect potential sensor faults among the swarm agents. The proposed methodology was tested in an experimental setup in the field in two marine testbeds: the Lagoon of Venice, Italy, and Biograd an Moru, Croatia. Achieved experimental results described in this paper validate and show the potential of the proposed approach.

Keywords: marine robotics; swarm; multi-vehicle system; underwater acoustic sensor network; cooperative control; underwater monitoring; anomaly detection

1. Introduction

As the importance of studying the impact of climate change and anthropogenic influences on both a global and local scale grows, novel technologies find new application niches. The ecosystem of the Lagoon of Venice, Italy is critical in a scientific, cultural, and socio-economic context. It also stands out as an area with pronounced interplay between global climate change-related effects and a variety of unique site-specific phenomena [1,2].

Eutrophication and the effect of intense human activity on various types of sediment and soils present in the lagoon interact with increasingly intense summer heatwaves and abundant storms and rainfall during other times of the year, leading to a pronounced presence of phenomena such as degradation of zooplankton, abnormal proliferation of certain species of macroalgae such as *Ulva rigida*, and hypoxic and anoxic events in the form of rapid localised (usually overnight) drops in the concentration of oxygen in the water. These hypoxic and anoxic events are becoming more frequent and more spatially widespread, and have been tied to fish mortality and biomass reduction and shifts, making them a significant object of study [3–6].

Long-term environmental monitoring in the waters of the Venice Lagoon is a significant logistical challenge, demanding a great number of boats, personnel, and other resources. The Horizon 2020

Future and Emerging Technologies project subCULTron was conceptualised as a novel technological answer to these issues, in the form of an autonomous heterogeneous swarm of marine robots, which, when working together, create a topologically reconfigurable underwater acoustic sensor network with surface access points [7,8]. The subCULTron underwater swarm can also be seen as a part of the Internet of Underwater Things (IoUT), which is roughly defined as "a network of smart interconnected underwater objects" [9].

Monitoring an environment using an underwater swarm consists of two key parts. The first part is *exploration*—the ability of the swarm to autonomously move throughout the environment of interest, discover new areas, measure, and map them. The second part is *anomaly detection*—the ability of the swarm to detect interesting areas in the environment while exploring. Exploration usually includes a strategy on how to determine the next location to be visited and explored. Anomaly detection is by definition the problem of finding patterns in data that do not conform to expected behaviour. In order for a swarm to be truly autonomous, both of these parts need to be tackled autonomously. Finding an anomaly (outlier) in collected data can mean that either (i) the swarm is detecting changes in the environment that are interesting for analysis or (ii) that one or more sensors are faulty. An example of the former case is the occurrence of hypoxia/anoxia, where the anomalous area has low/no oxygen concentration measurements, in contrast to higher values present in the areas surrounding it.

Monitoring in underwater environments poses a set of unique challenges. While underwater, communication, which is the backbone of every swarm algorithm, is limited to low-bandwidth acoustic signals which are sensitive to signal interference, resulting in potential communication package loss. Since working with swarms means dealing with a large number of robotic units, it is also realistic to assume that these units are resource-constrained (limited processing power, storage, energy capacity), hence, any algorithms used should by necessity be of lower complexity. Since long-term monitoring involves a high probability of equipment damage and wear, none of the units should be a central point of failure, and there should be a method available for detection of faulty units. All of these factors lead to the conclusion that a distributed solution capable of fault-detection while operating online is necessary—and it is this solution that the subCULTron swarm aims to provide.

Several approaches to monitoring an underwater environment using a multi-robot system exist in the literature. One potential solution is using a heterogeneous system, such as those given in [10,11], where the authors use aerial, surface, and underwater robots, and deploy a system for gathering multi-domain data at coral reefs. The authors give a monitoring timeline with distinctive phases. A frequently seen approach, such as in [12,13], is using Lagrangian (current-following) drifters, capable of vertical movement.

In [14], the authors use an underwater sensor actuator network to detect extreme temperature gradients in a one-dimensional vertical setting. They propose an adaptive sampling algorithm based on binary search for the network to reliably detect a thermocline. The method is validated in a simple laboratory setting with mock-up sensor nodes and the authors do not consider potential sensor faults.

Systems similar to the one described in this paper have been developed for long-term underwater monitoring of coral reefs and fisheries [15] or oil exploration [16]. These approaches utilise fixed underwater sensors collecting data about the environment and Autonomous Underwater Vehicles (AUVs) acting as communication relays or data hubs tasked with the delivery of aggregated data to the surface. The system described in [15] employs two modes of communication (short-range optical and long-range acoustic) to better utilise the available energy on the sensing unit, thus increasing deployment longevity. The subCULTron system differs from the above-mentioned approaches by equipping its underwater sensor hubs with z-axis movement capabilities. This enables autonomous reconfiguration of the sensor network during mission execution either by utilising currents for drifting or using Autonomous Surface Vehicles (ASVs) as aids for relocation.

Exploration is a well-studied topic for ground and aerial vehicles, where most approaches consist of detecting the frontiers of the explored area and then estimating the utility of visiting each frontier point next. Utility balances information gain, travelling costs, and localisation ability for each point [17]. In an underwater setting, there are fewer approaches, focusing mostly on specific hardware properties [12,13,18] or exploration strategies [19]. Recently, in [20], the authors used information-theoretic exploration and sampling together with ocean models to derive an exploration strategy. An approach is given for a single-vehicle case, but a similar method could be adapted for a multi-robot setting as well.

An extensive overview of anomaly detection algorithms can be found in [21]. Only recently have truly distributed solutions started to emerge. These include consensus algorithms, such as trust-based consensus [22] and median consensus [23,24]. Agents agree on the value of interest (trust, median of all measurements) and outlier candidates are detected based on this value. Some approaches use movement capabilities of robots acting as sensor hubs for detection and tracing of anomalies [25] and neural network-based detection [26]. Similar to consensus, distributed Kalman filters can be used for state estimation and determination of outlier candidates through analysis of covariance matrices [27–29].

In this paper, exploration as a concept is only briefly discussed and the emphasis is instead placed on how the subCULTron swarm is designed to be continually moving through the environment and reaching new areas. In particular, strategies for deciding where to go next are not studied, as this topic requires extensive consideration and is a separate promising research direction.

The focus of the paper and the main contribution is a paradigm for distributed detection and verification of anomalies in the environment using a heterogeneous marine robot swarm. As opposed to other anomaly detection algorithms that operate on a given static data set, the motion capabilities of robot units and the reconfigurability of the swarm are exploited to collect additional data and verify the nature of the initially-obtained anomalies—or detect potential sensor fault among the swarm agents. As an important real-world application of the swarm is detecting and monitoring the anoxia phenomenon in the Lagoon of Venice, this proposed paradigm was tested in the field using an analogous proof-of-concept experimental scenario. Even though the approach is described as implemented and used by the subCULTron system, concepts can be transferred to other robotic systems as well.

The paper is structured as follows. Section 2, Materials and Methods, contains a description of the subCULTron robotic system Section 2.1 and an outline of the proposed paradigm and resulting experimental scenario Section 2.2. The experimental setup and results are given and discussed in Section 3. The final section contains a conclusion and directions for future work.

2. Materials and Methods

2.1. Subcultron System Description

The subCULTron multi-agent system [30] was envisioned as an artificial marine ecosystem, aiming to reduce the need for boats and personnel manually deploying and collecting measuring instruments at sea by automating the deployment, reallocation, and collection of monitoring devices, as well as data acquisition. Besides its autonomy, this system has another significant advantage inherent to its large structure, which is the ability to measure environmental factors from multiple locations simultaneously and to reconfigure its topology dynamically. The system is comprised of three layers: five Autonomous Surface Vehicles (ASV) called aPads (artificial lily pads) constitute the top layer, a small swarm of aFish (artificial fish) represent the middle layer, and more than 100 underwater sensor nodes called aMussels (artificial mussels) act as the bottom layer. The monitoring paradigm proposed in this paper makes use of the aPad and aMussel agent types. A detailed description of the functionalities and agent interactions featuring the aMussel and aPad robots (Figure 1), as well as the subCULTron system as a whole, is presented in [31]. Here, an overview is given to provide necessary context.

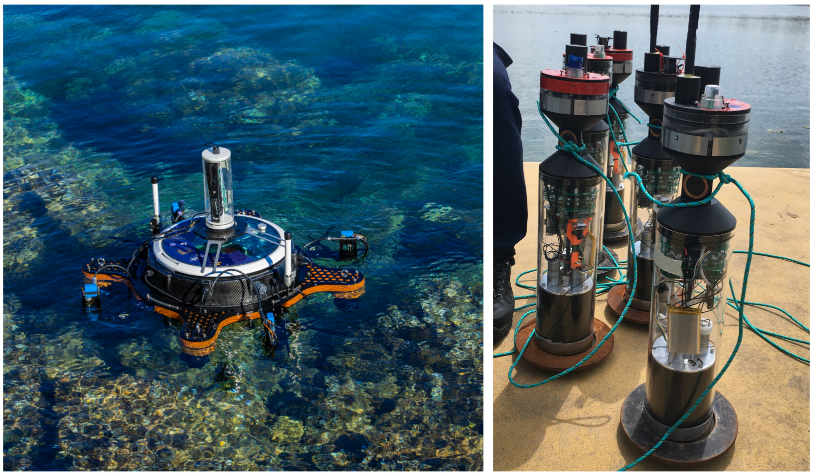


Figure 1. aPad (**left**) and aMussel (**right**) robots.

2.1.1. AMussel

The aMussel agents serve as low cost underwater perception hubs, working together to establish a sensor network for distributed environmental monitoring and long-term data collection. As the primary measurement devices, they house a variety of sensors capable of monitoring relevant environmental factors. The design of the aMussel is represented in Figure 2. For the application described in this paper, the oxygen concentration, pressure, temperature, and turbidity sensors were installed, but any other compatible sensors can easily be added due to the robot's modular design. For long-range communication, aMussels rely on miniature acoustic modems called nanomodems while submerged, and have Bluetooth, GSM, and WiFi capabilities when on the surface.

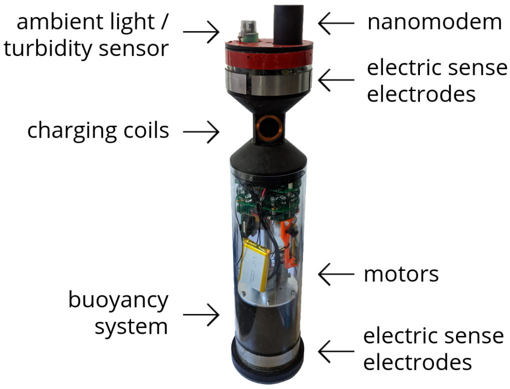

ambient light / turbidity sensor
nanomodem
electric sense electrodes
charging coils
motors
buoyancy system
electric sense electrodes

Figure 2. The aMussel robot—sensors, actuators and communication devices.

The aMussel was developed with long-term autonomy in mind, which is imperative for successful monitoring of environmental processes with slow dynamics. Its main electronic board, called the MU (Measurement Unit) board, is capable of deep hibernation, which drastically reduces energy expenditure. The rest of the modules connected to the central board can be independently powered off from the central unit, providing additional energy saving. During its operation, an aMussel may switch between modes of sleep and active perception and communication, depending on the application scenario. Each aMussel unit also carries a Raspberry Pi board with a custom-made adapter board, powered on in short intervals to provide additional processing power and data storage. To further prolong the operation of aMussel agents, a unique docking mechanism with inductive coils was devised, using which an aPad can grasp the aMussel and charge it (Figure 3). Recent related work [32] regards the analysis of long-term autonomy of the system, mainly from the standpoint of energy exchange between aPad and aMussel robots. The paper contains an examination of data acquired during efforts to model subCULTron swarm agent batteries as well as validate the long-term potential of the developed system. Of particular interest, is a single aMussel battery discharging with the MU board turned on but idle, with approximately 16 h needed for full discharge, and the aMussel battery discharging with the MU board in sleep mode, waking up once every hour for about 10 s in order to record measurements from all sensors, where 10 days were needed for full discharge. While taking into account the batteries' ageing and ability to hold charge [33], switching to and actively using the aMussel's identical secondary battery after its primary has been depleted effectively doubles this total deployment time. Since the energy exchange abilities of the agents greatly impact the longevity of the aMussel portion of the swarm, several cases of the aMussels recharging their batteries were also recorded. Here, of particular interest was the aMussel primary battery charging with the MU board turned on but idle (approximately 9.5 h needed for full charge), the aMussel primary battery charging with sleep, waking up briefly in regular intervals to record sensor data (6.5 h), and the aMussel secondary battery charging with the same sleep behaviour, with the MU powered by the primary battery (8 h). For comparison, battery data of an aPad running its on-board computer and Kinect camera and activating all of its thrusters for 3 s every 30 s was recorded, leading to a result of complete discharge after 27 h.

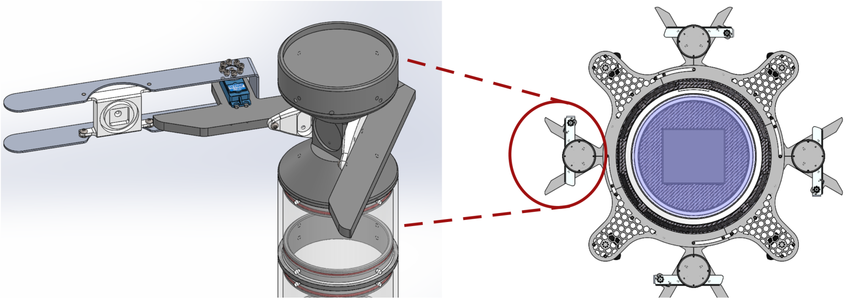


Figure 3. aPad docking mechanism design. Details show an aMussel top cap being grasped.

Since aMussels are mainly concerned with collecting data while resting at the seabed, their design involves limited movement capability, with a buoyancy system which allows them to sink to the seafloor, float to the surface, or hover at a certain depth being their sole actuator. They rely on assistance from the aPads for any more complex movement: when surfaced, the aMussel can be grasped and transported by an aPad, with the same docking mechanism used for energy transfer (Figure 3).

The aMussel user software runs on the MU Board with a Cypress PSoC4 (Programmable System-on-Chip) microcontroller running FreeRTOS and Embedded C software. PSoC programming tools allow easy-to-use graphical software redesign when changes in hardware occur. A high-level Application Programming Interface (API) written in C++ enables secure and reliable access to all sensors and actuators and keeps implementation details of device management hidden away from algorithm designers.

2.1.2. APad

The aPad, pictured left in Figure 1, is an overactuated Autonomous Surface Vehicle (ASV). The primary role of the aPad in the swarm is to aid and assist underwater units, primarily by supplying energy and movement capability. They also serve as a link to any potential human observer and operator, as they take part in both the underwater and surface communication network. Below the surface, aPads communicate using the same nanomodem devices that are present on the aMussel units. For surface communication purposes, they have a mesh-capable wireless router which provides both WiFi access points for aMussels and surface stations and a mesh network for the aPads themselves.

The platform has four thrusters in an X-shaped configuration which make it overactuated and omnidirectional. Each aPad has four mechanical docking stations with inductive transmitter coils, and can charge up to four aMussels at a time. To enable autonomous grasping of surfaced aMussels, besides the mechanical docking station design, a visual servoing system was devised comprising a Kinect sensor and a specially designed pan mechanism [34].

Another vital role of the aPads is one of anchors in the process of underwater localisation [31]. In underwater sensor localisation, anchors positioned at known positions provide the source of information necessary for the localisation of unlocalised underwater nodes. Underwater nodes then collect measurements to gather information about the ranges between them and anchors. Then, using these measurements and simple algorithms such as multilateration, the unknown node positions can be calculated. For their own localisation and positioning, each aPad has an IMU and a GPS module. With its localisation and navigation capabilities, the aPad can return to a home position and have its batteries recharged while deployed thanks to a waterproof charging jack on its hull.

The aPad software runs on an Intel NUC mini PC, its main on-board computer. Similarly to the aMussel, the code is structured hierarchically and provides end-users with a simple API written in the Python programming language, enabling high-level mission planning and execution. Lower-level controllers, thruster allocation, sensor drivers and utilities, and navigation filters are implemented in C++ and Python and organised in packages under the Robot Operating System (ROS) paradigm.

2.2. Formal System Description

This section describes the proposed paradigm for underwater monitoring. The goal of the system is to find and confirm anomalies in a certain area of the seabed. The first step is for the aPads to deploy a group of aMussels, called the exploration group, to different predefined locations. After deployment, this group sinks to the seabed and begins an acoustic exchange of measurements. If one aMussel detects anomalous measurements, it becomes an outlier candidate. In order to validate the outlier candidate, a new group of aMussels, called the verification group, is deployed by aPads to its position. The verification group of aMussels exchanges measurements with the outlier candidate and either confirms or disproves that the area to which they have been deployed is anomalous. The final step includes the relocation of the outlier candidate, either to a home position for repair if the outlier is disproved, or to a new location to explore if the anomaly is confirmed and hence the outlier agent is “trusted”.

Due to the low cost and high number of the aMussel units, it is justified to use a certain number of them for verification of potential anomalies, as well as for future monitoring of a confirmed anomaly. In addition, it is possible to dynamically reallocate aMussels from the verification group to the exploration group in cases where there is a need to explore larger areas.

The subCULTron swarm used in in this paper consists of two types of robotic agents: aPads $\mathbf{P} = \{P_1, P_2, P_3, ..., P_n\}$ and aMussels $\mathbf{M} = \{M_1, M_2, M_3, ...M_m\}$, where n is the number of aPads used and m is the number of aMussels used.

The state of each aPad P_i in time step k can be described with:

$$P_i^k = (q_{pi}^k, \mathbf{d_i^k}) \tag{1}$$

where $q_{pi}^k = (x_{pi}^k, y_{pi}^k)$ represents aPad GPS coordinates and $\mathbf{d_i^k} \in N_0^4$ is the status of each aPad’s four docks. Element d_{ij}^k in $\mathbf{d_i^k}$ assumes a value of 0 if the dock is empty or a value in range $[1, m]$ which corresponds to the ID number of the aMussel in dock j.

The state of each aMussel M_i in step k can be described with:

$$M_i^k = (q_{mi}^k, v_i^k, \mathbf{s_i^k}) \tag{2}$$

where $q^k_{mi} = (x^k_{mi}, y^k_{mi})$ represents aMussel GPS coordinates, v^k_i is a sensor measurement value and $\mathbf{s^k_i} = (s^k_i, c^k_i, o^k_i)$ is the aMussel status vector. s^k_i assumes value 1 if the aMussel is on the surface and value 0 if the aMussel is on the seabed, c^k_i assumes value 1 if the aMussel is in an aPad dock and 0 otherwise, and o^k_i contains information on whether the aMussel is considered to be an outlier.

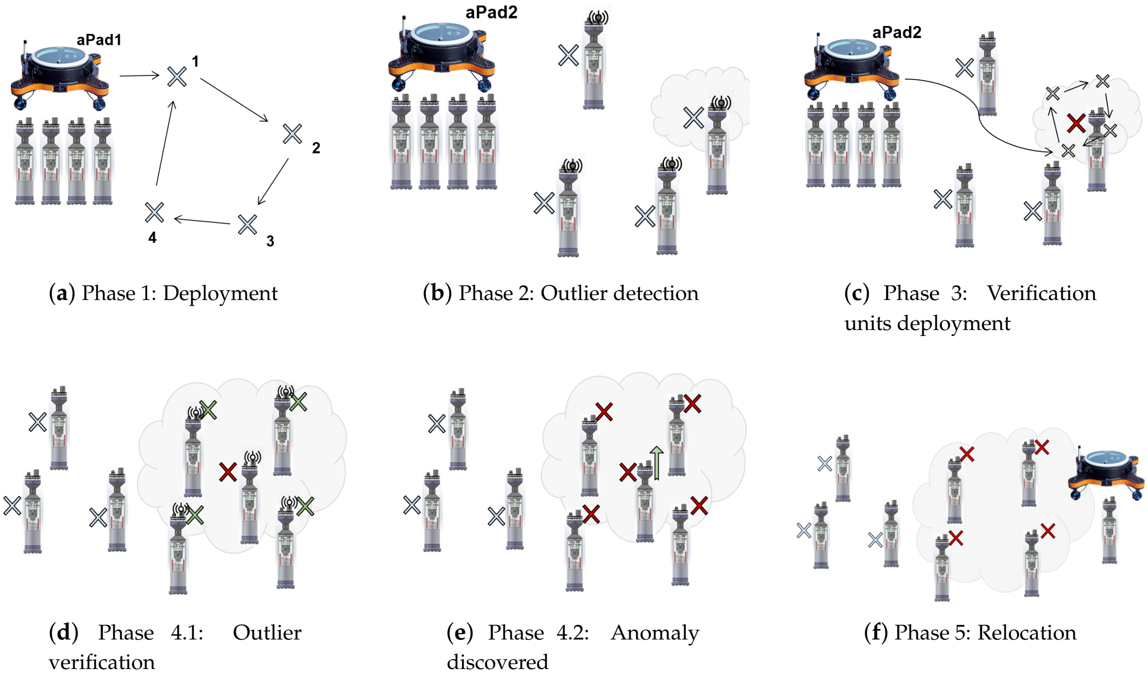
During the experiments, it is assumed that aMussel coordinates q^k_{mi} correspond to the coordinates where it was released by the aPad. The information about s^k_i is determined from the position of the aMussel's buoyancy piston, while the value of c^k_i can be determined from the aMussel's charging module which will return a value of 1 when the aMussel is connected to the aPad dock charger.

M can be presented as a union of different groups of aMussels:

$$\mathbf{M} = \mathbf{M_e} \cup \mathbf{M_{v1}} \cup ... \cup \mathbf{M_{vr}} \tag{3}$$

where $\mathbf{M_e} = \{M_{e1}, ..., M_{em_e}\}$ is a group of aMussels committed to exploration and $\mathbf{M_{vi}} = \{M_{vi1}, ..., M_{vim_{vi}}\}$ is one of r groups of aMussels committed to verification. The number of elements in $\mathbf{M_e}$ is m_e, and number of elements in $\mathbf{M_{vi}}$ is m_{vi}, where $m_e + m_{v1} + ... + m_{vr} = m$. The acoustic communication channel is such that every aMussel can hear every other aMussel in range, but they are programmed to ignore messages sent by aMussels from any group other than their own. In the experiments presented in this paper, only one verification group is used.

As mentioned earlier, finding and confirming anomalies in a certain area of the seabed is the end goal of the proposed paradigm. Initially, all aMussels are docked to the aPads. The first step is to deploy aMussels in the exploration group $\mathbf{M_e}$ over the chosen area and, in case the anomalous position is found in that area (by recognising the measurement of one aMussel as an outlier), an attempt is made to verify it by one of the verification groups of aMussels $\mathbf{M_{vl}}$. For each anomalous position found by the exploration group, one verification group is deployed. The experiment comprises several phases, which are described in the continuation and shown together in Figure 4.



(**a**) Phase 1: Deployment

(**b**) Phase 2: Outlier detection

(**c**) Phase 3: Verification units deployment

(**d**) Phase 4.1: Outlier verification

(**e**) Phase 4.2: Anomaly discovered

(**f**) Phase 5: Relocation

Figure 4. A graphical overview of the phases of the environmental monitoring scenario. (**a**) Exploration units deployment (**b**) Outlier detection (**c**) Verification units deployment (**d**,**e**) Outlier verification (**f**) Relocation.

2.2.1. Exploration Units Deployment

The first step in the experiment includes the deployment of the $\mathbf{M_e}$ group to a predefined set of GPS coordinates $\mathbf{q_d} = \{q_{d1}, ..., q_{dt}\}$. This is achieved using the procedure described in Algorithm 1. In general, the number of predefined GPS coordinates t will be greater than or equal to m_e, the number of aMussels in the exploration group.

Algorithm 1: Deployment procedure

```
Data: $\mathbf{P}$, $\mathbf{M_e}$, $\mathbf{q_d}$
Result: aMussels in $\mathbf{M_e}$ deployed to $\mathbf{q_d}$
e:=1;
for $i$=1 to $n$ do
    for $j$=1 to 4 do
        if $M_{d_{ij}}$ is in $\mathbf{M_e}$ then
            Move $P_i$ to $q_{de}$;
            $P_i$ release aMussel $M_{d_{ij}}$;
            e:=e+1;
        end
    end
end
```

During the deployment procedure, the aPad P_i with an exploration group aMussel $M_{d_{ij}}$ goes to the location q_{de}, and opens the dock containing $M_{d_{ij}}$, which releases it. This procedure is repeated for all aMussels in the exploration group.

aMussels are programmed so that on the falling edge of c_i^k (which represents the moment of release from the aPad dock) they initiate the sinking procedure. The state machine of an aMussel in the exploration group is shown in Figure 5. Once again, since the aMussels start sinking immediately upon their release from the aPad dock, it is assumed that their GPS coordinates on the seabed (q_{mi}^k) correspond to the coordinates where the aPad released them (q_{de}) and are as such recorded by the deploying aPad. In use-cases where significant depth and strong currents are present in the target environment, acoustic localisation with aPads as surface anchors can be included in the algorithm.

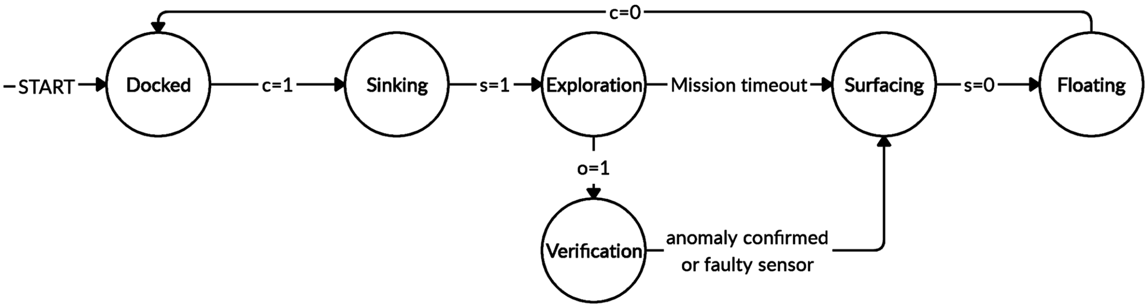


Figure 5. State machine of an aMussel in the exploration group.

For the experiment presented in this paper, this step includes the aPad designated as P_1 deploying four aMussels to predefined positions, where they sink to the seabed (Figure 4a). The experimental scenario for this aPad is shown in Figure 6.

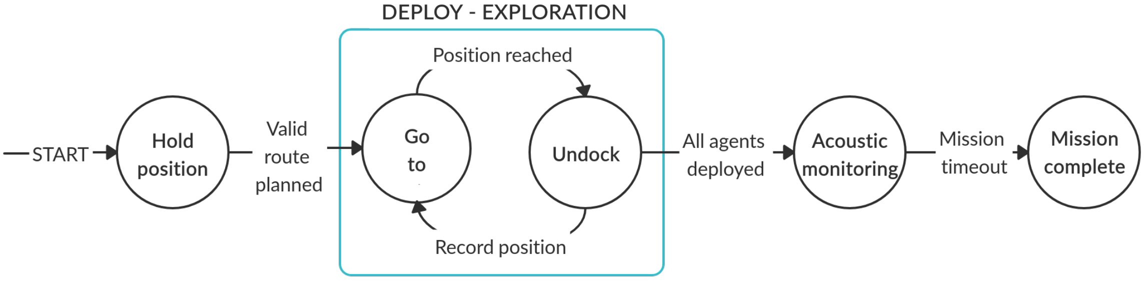


Figure 6. Anoxia scenario experiment for the aPad deploying exploration group agents (aPad P_1).

2.2.2. Outlier Detection

After the aMussels from exploration group $\mathbf{M_e}$ have been deployed to the designated coordinates, they start to exchange their current sensor measurement values v_i^k via acoustic link, in an effort to detect any outlier in the measurements. Different algorithms for outlier detection can be used for this purpose.

For demonstration purposes, in the experiment described in this paper, a simple outlier detection algorithm was used which assumes complete communication topology between all aMussels in the group. Since all aMussels receive measurements from the rest of the group, it is possible to compute a group average. In the case when the largest individual deviation from the average value is greater than some predefined value δ, it is assumed that an outlier has been found. The choice of δ is determined heuristically, based on the expected range of the observed measurements, as well as on the value of the measurement deviation that can be considered an anomaly.

The group average in step k is calculated as:

$$Avg^k = \frac{\sum_{i=1}^{m_e} v_{ei}^k}{m_e} \tag{4}$$

The maximal deviation from the average is:

$$\Delta_{max}^k = \max_{i \in [1, m_e]} |v_{ei}^k - Avg^k| \tag{5}$$

where Δ_{max}^k is the maximal deviation from the group average in step k.

If the condition:

$$\Delta_{max}^k > \delta \tag{6}$$

is satisfied, then the argument o of the outlier aMussel M_o is:

$$o = \arg \max_{i \in [1, m_e]} |v_{ei}^k - Avg^k| \tag{7}$$

The algorithm for outlier detection is presented in Algorithm 2. Data exchange between aMussels is contained in the subfuction *CalculateAverage*, where each aMussel shares its measurements (v_i^k) with the rest of the group and calculates the average based on the received measurements from other aMussels in the group.

Algorithm 2: Used outlier detection algorithm

```
Data: $v_i^k$
Parameter: $\delta$
Result: outlier / outlier not found
avg:=CalculateAverage($v_i^k$ in $\mathbf{M_e}$);
for $i$=1 to $m_e$ do
    if $i$==1 or abs(avg-$v_i^k$)>maxDeviation then
        maxDeviation:=abs(avg-$v_i^k$);
        possibleOutlier:=i;
    end
end
if maxDeviation>$\delta$ then
    outlier:=possibleOutlier;
else
    outlier not found
end
```

If the outlier is identified in some unit M_o, this unit adds the information that it is an outlier to its broadcast acoustic message. This broadcast designates that M_o changes its group from $\mathbf{M_e}$ to the first available verification group $\mathbf{M_{vl}}$, so the rest of the $\mathbf{M_e}$ group starts ignoring messages from unit M_o. This broadcast also triggers the deployment of units that belong to the group $\mathbf{M_{vl}}$.

The implemented underwater acoustic communication is based on a time scheduling protocol using the widely-used round-robin principle presented in [31]. Time slots are assigned equally among all agents in a circular order, handling communication without priority. Each of the agents in the swarm has an equal amount of time for data sharing, which is well suited to periodically reporting a desired measured environmental variable.

The described simple outlier detection algorithm can be replaced with other outlier detection algorithms, as long as the algorithm remains distributed and suitable for operation in an underwater setting. As mentioned in the Introduction, promising approaches tested in underwater settings are, for example, median-based consensus [24] and trust-based consensus [22].

2.2.3. Verification Units Deployment

Similar to during exploration units deployment, the goal of verification units deployment is for aPads to deploy a group of aMussels to the designated coordinates, where they sink to the seabed.

Once the outlier M_o has been identified, the first available verification aMussel group $\mathbf{M_{vl}}$ is deployed to positions around the outlier q_{mo} using an algorithm similar to Algorithm 1, but this time for the $\mathbf{M_{vl}}$ group. The purpose of the verification units is to verify whether the identified outlier aMussel has correct sensor readings or if it is faulty.

For the experiment presented in this paper, this step includes the aPad designated as P_2 deploying four aMussels to positions around q_{mo} (see Figure 4c). The experimental scenario for this aPad is shown in Figure 7.

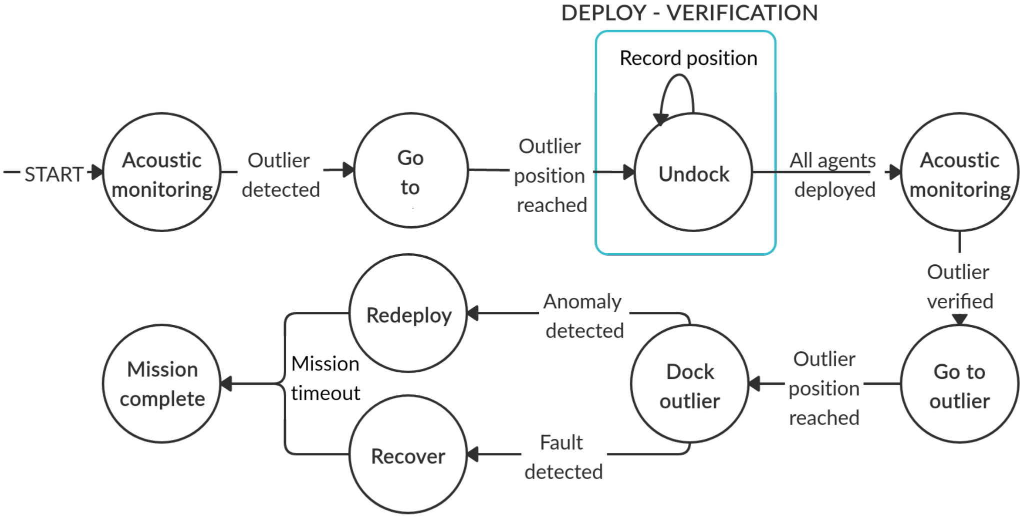


Figure 7. Anoxia scenario experiment for the aPad deploying verification group agents (aPad P_2).

2.2.4. Outlier Verification

With the arrival of the aMussel group $\mathbf{M_{vl}}$ to the location of the outlier M_o, which is now part of the $\mathbf{M_{vl}}$ group, they start to exchange measurements in an effort to identify any potential outlier among themselves. Since all aMussels in $\mathbf{M_{vl}}$ group are located at the same coordinates, if all of them are functioning properly, their measurements should coincide. The same outlier detection algorithm as previously, presented with Equations (4)–(6), is used. In the case when no outlier is found in the group $\mathbf{M_{vl}}$, it is considered confirmed that aMussel M_o has found the sought-after environmental anomaly, since the rest of the group $\mathbf{M_{vl}}$ has confirmed its sensor readings. In a case where M_o is determined to be the outlier once again, it can be assumed there is something wrong with its sensors and it is presumed faulty.

2.2.5. Relocation

In the event that the outlier is verified and the anomaly detection is confirmed, M_o broadcasts a message that it is a "trusted" unit with functioning sensors. This broadcast signals to the aPads that it is ready for reallocation to the next exploration point. It also signals that M_o is again moved from the $\mathbf{M_{vl}}$ group back to the $\mathbf{M_e}$ group. After the M_o agent surfaces, the aPad docks it autonomously and brings it to a new location for exploration (the "Redeploy" step shown in Figure 7). The rest of the $\mathbf{M_{vl}}$ group is left to monitor the anomaly.

In case of confirmed sensor fault in M_o, it again surfaces to be picked up by an aPad, but in this case it is moved to a predefined home position to await repair (the "Recover" step shown in Figure 7). In this case, the group $\mathbf{M_{vl}}$ becomes part of the exploration group $\mathbf{M_e}$ and continues to monitor the environment.

In both cases, the relocation phase includes the arrival of the aPad to the coordinates of the aMussel M_o, which triggers the M_o surfacing—when, on the surface, the aPad autonomously docks the aMussel and carries it to the selected location.

3. Results

The results of the proof-of-concept field experiments conducted with the swarm in order to test the viability of the proposed paradigm are presented and discussed in this section. The experiment scenario was designed around the concept of detecting and monitoring the anoxia phenomenon in the Lagoon of Venice.

3.1. Setup

Proof-of-concept field experiments were conducted in the Adriatic Sea, in Biograd na Moru, Croatia, and Venice, Italy. The first testbed was an outdoor pool/walled off section of the sea providing a known environment for initial algorithm validation, while the second presented a "stress test" in a challenging environment on-site near Sant'Angelo della Polvere in the lagoon of Venice, Italy. In both cases, to ensure repeatability as well as to gather valuable data, the experiments took place over the span of a week, with the experimental scenario running three times a day for five days. Two examples of experiments in progress showing the swarm agents in different marine testbeds are shown in Figure 8.

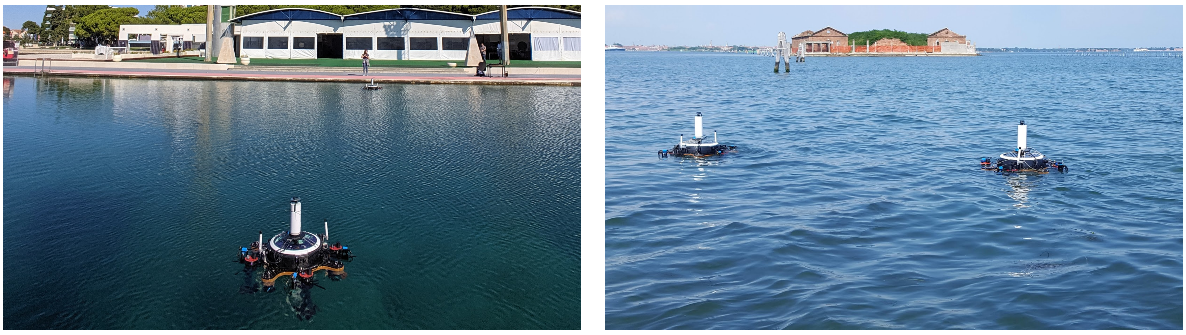


Figure 8. Experimental area in Biograd na Moru, Croatia (**left**); experimental area in the Venice Lagoon (**right**).

The experiments in both testbeds were carried out by two aPad platforms and eight aMussel sensor nodes (four per aPad for full capacity, forming two groups—one for exploration and one for verification). Measurements from the aMussels' pressure sensors were chosen as the relevant data in the proof-of-concept experiments, due to the ease of getting consistent and predictable pressure measurements of similar values in the given testbeds—in the target application of anoxia monitoring, the readings from the aMussels' oxygen sensors would be used (Figure 9 presents the example of oxygen concentration measurements taken by aMussels over a period of four nights in the Lagoon of Venice, where a number of agents were left overnight on the seabed to monitor the oxygen levels, clearly showing nightly drops). One of the aMussels from the first group was programmed as the designated outlier, i.e., it had an offset applied to its pressure sensor readings compared to other agents in the initial exploration group. For experiments where the outlier aMussel had located an environmental anomaly, the second group of aMussels had this same offset applied to it. For experiments where the outlier aMussel had suffered some sort of fault or failure, no other aMussels had the offset applied. During both experiments, the value of δ was set to 100 hPa.

Four starting aMussel deployment positions were chosen within each of the experimental areas. These aMussel positions are shown in Figure 10. Repeated experiments were performed using the same starting aMussel position configurations for their respective experimental areas; however, various physical aMussel units were used interchangeably. The experiment scenarios were started on the robotic agents by human operators, and from there continued entirely autonomously.

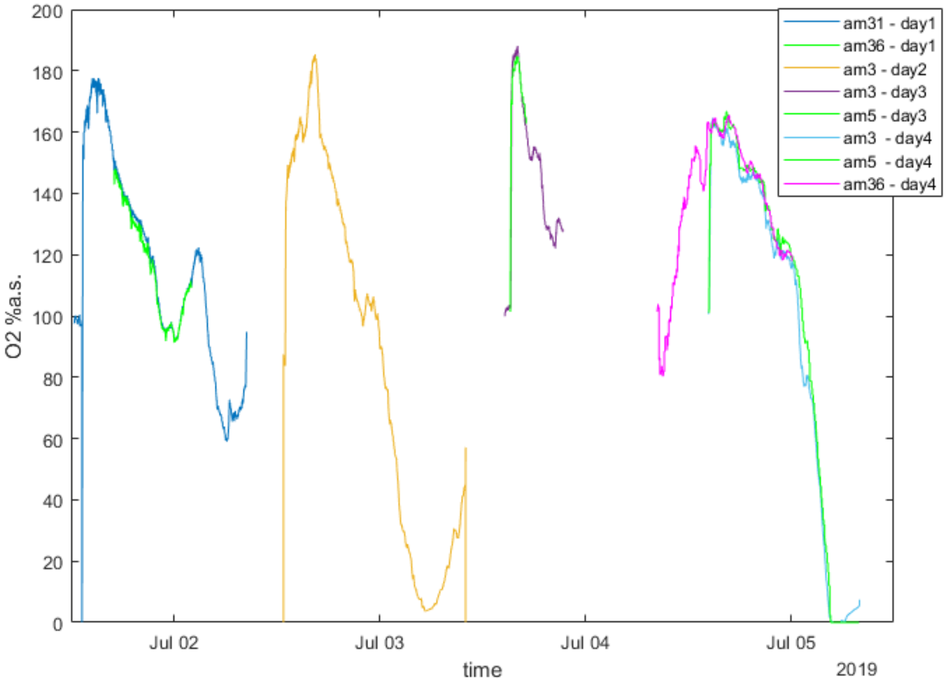


Figure 9. Example of oxygen data measured by aMussels.

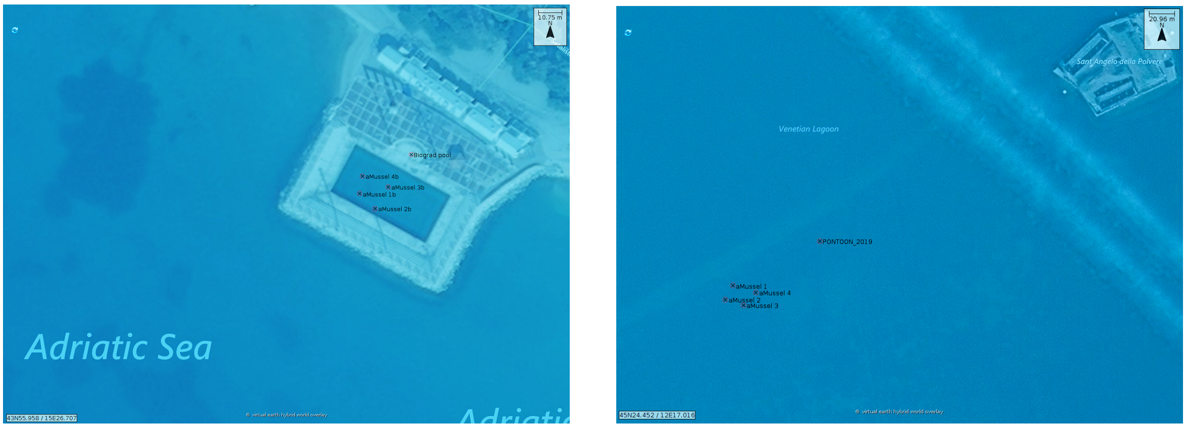


Figure 10. The two experimental areas: initial selected aMussel positions used in all experiments, overlaid on map with satellite image. Biograd na Moru, Croatia (**left**); Venetian Lagoon (**right**).

3.2. Experimental Results

This section contains a detailed overview of results achieved during the proof-of-concept experiment conducted for validation purposes in an outdoor pool in Biograd na Moru, including a mission replay reconstructed from data logged on all the vehicles during the duration of the experiment. A video containing an animated showcase of this data combined with several recordings made using a drone and underwater cameras are available at http://www.youtube.com/watch?v=d4_9WKasei0.

The main points of a mission replay are shown in the following segment, illustrated by images reconstructed from real data at key points during the experiment scenario. Figure 11 depicts the initial state of the system, with two aPads holding their position, at full aMussel carrying capacity. The experiment proper then begins by aPad P_1 deploying its docked aMussel group $\mathbf{M_e}$ containing aMussels 11, 14, 33, and 34 (marked in blue) to four predefined positions. The aMussel which was programmed to be the outlier in the experiment—aMussel 11—is shown in red.

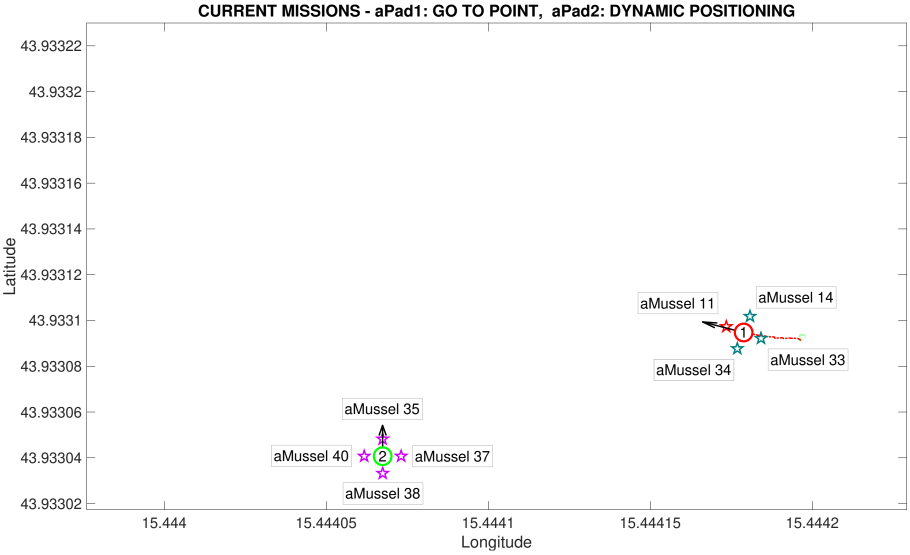


Figure 11. Initial deployment started in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

Figure 12 shows the state after the first aMussel group has been deployed: aPad P_1 is set to hold its position near the final deployment, as all four deployed aMussels sink to the seabed where they start collecting measurements and exchanging collected information amongst themselves via acoustic communication. The aPad trajectories are plotted with colours corresponding to the mission primitive the aPad was executing at the time—go to point (red), dynamic positioning (green), or docking/undocking (blue). The labels next to each aMussel on the mission replay show the pressure data transmitted acoustically. Each aMussel broadcasts its sensor data to all other acoustic communication-capable agents in the system in a round-robin fashion, and the sending of each of these acoustic packets is indicated in the mission replay by the respective aMussel label updating and turning green, as seen in Figure 13. This marks the start of the outlier detection phase.

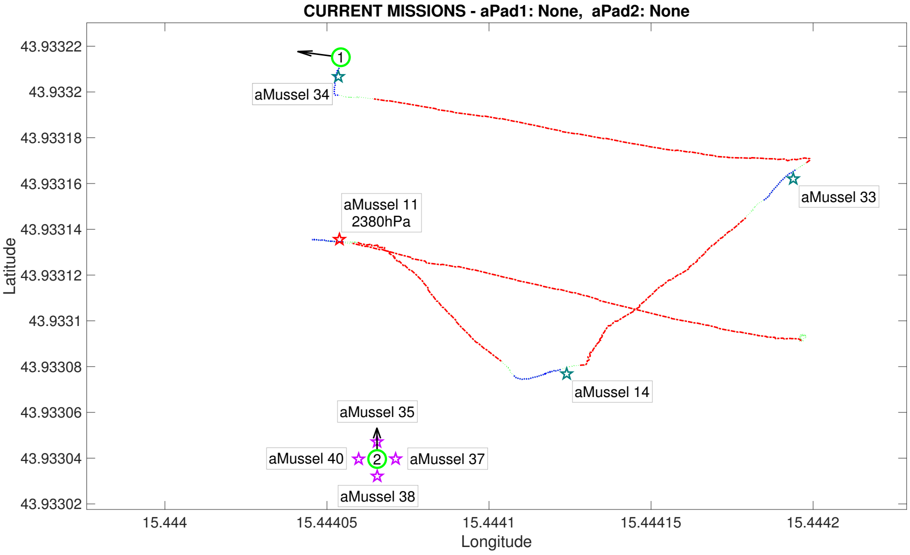


Figure 12. Initial deployment endeded in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

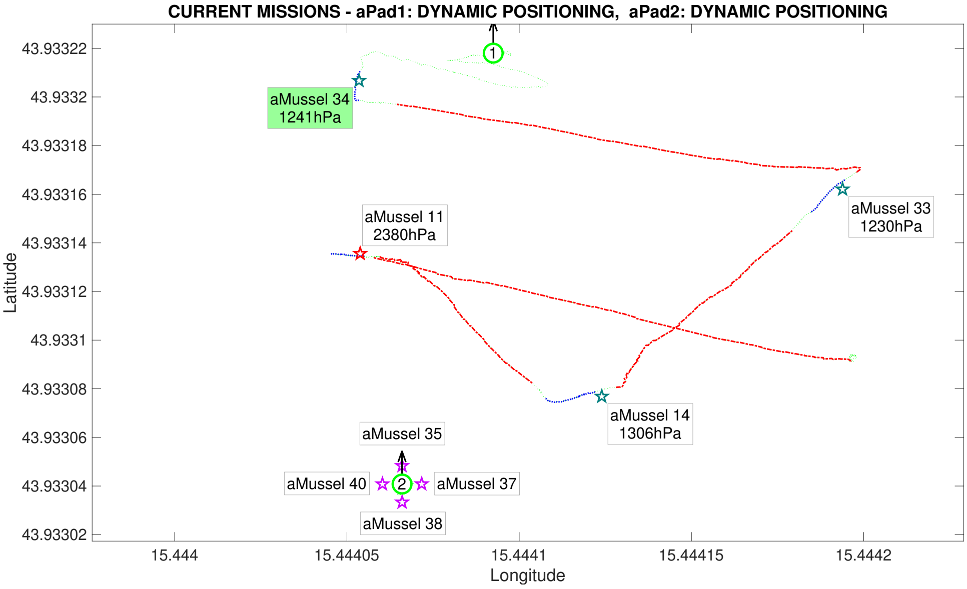


Figure 13. Outlier detection started in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

After exchanging a sufficient amount of data to reach a consensus, the aMussels decide if there is an outlier among them. In the event of a successful detection, the outlier aMussel (which is, as part of the exploration group, "aware" of its own outlier status) transmits the acoustic message that is the outlier to the aPads. Figure 14 shows the outlier aMussel label in orange, the agent having just reported its status, with the final pressure sensor data showing its measurements as over 1000 hPa higher than its peers.

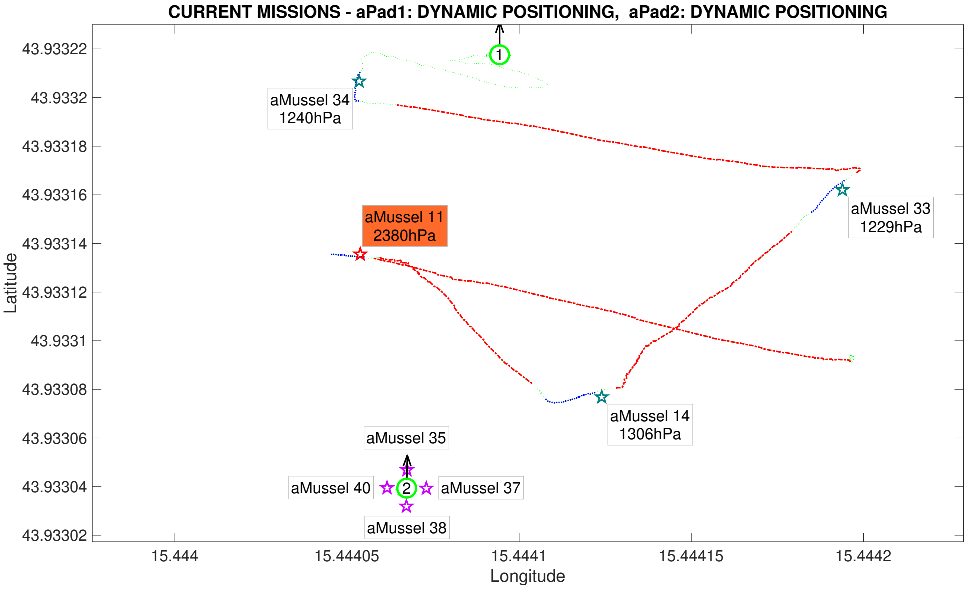


Figure 14. Outlier detection completed in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

The next phase of the experiment is verification whether the outlier aMussel is reporting on an actual environmental anomaly, or if its sensor is faulty. In this phase, aPad P_2 deploys the aMussel group $\mathbf{M_{vl}}$ containing aMussels 35, 37, 38, and 40 (marked in purple) in the vicinity of the outlier aMussel to confirm or refute its measurements, as can be seen in Figures 15 and 16.

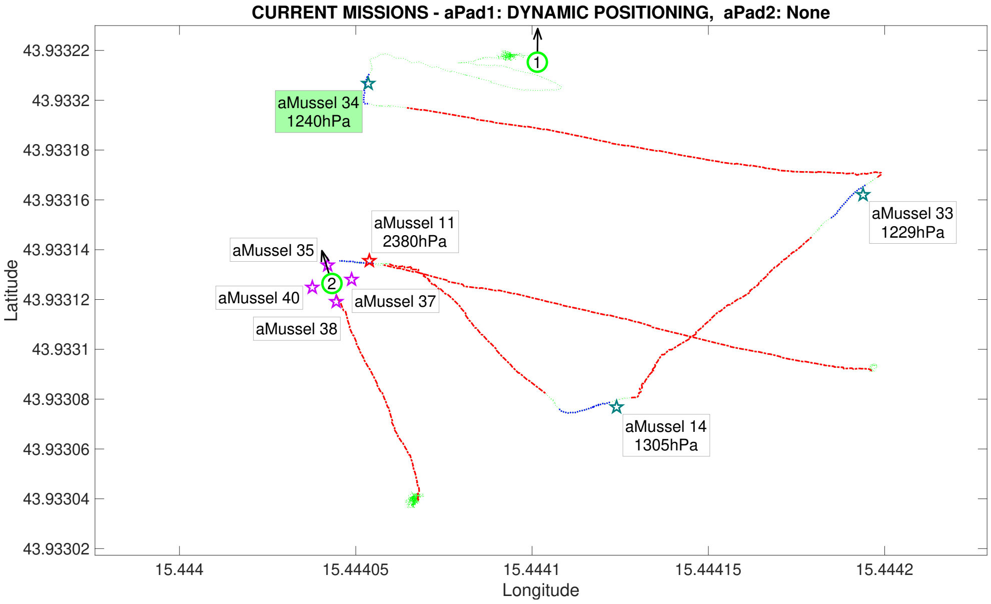


Figure 15. Second deployment started in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

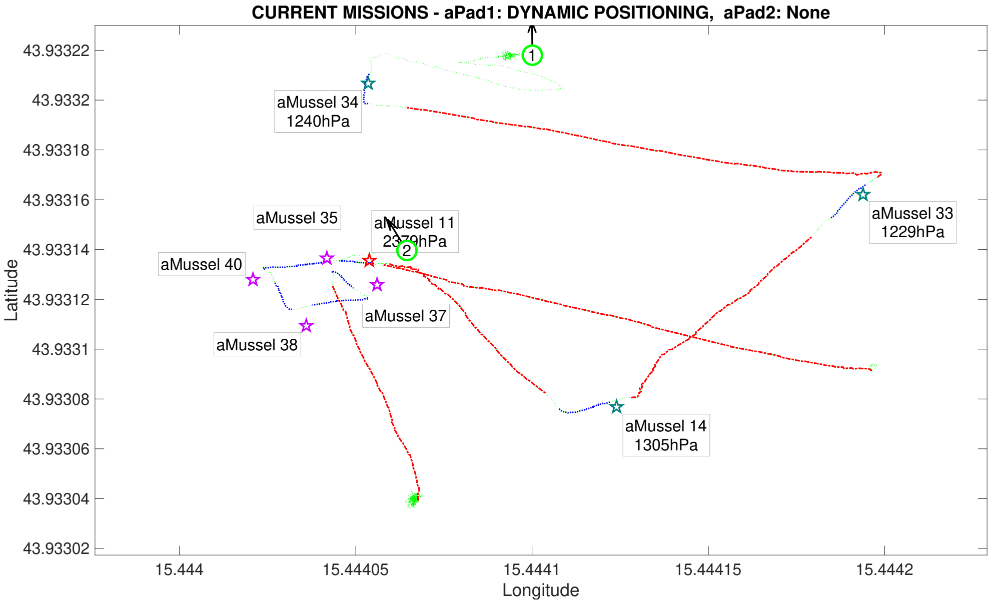


Figure 16. Second deployment completed in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

Another instance of outlier detection is started, by having the aMussels comprising group $\mathbf{M_{vl}}$ and the original outlier exchange pressure sensor data acoustically, disregarding the data points being transmitted by the aMussels from group $\mathbf{M_e}$ (Figure 17).

In this particular instance of the experiment, the assumption was that the designated outlier aMussel had indeed found an anomaly, meaning that the second group of deployed aMussels had the same programmed sensor value offset, which can be seen in their data labels. Once consensus has been reached, the now verified outlier aMussel sends a message to the aPads requesting relocation (represented by its cyan label), and the nearest available aPad moves to its position in order to collect it for further redeployment and exploration, shown in Figure 18.

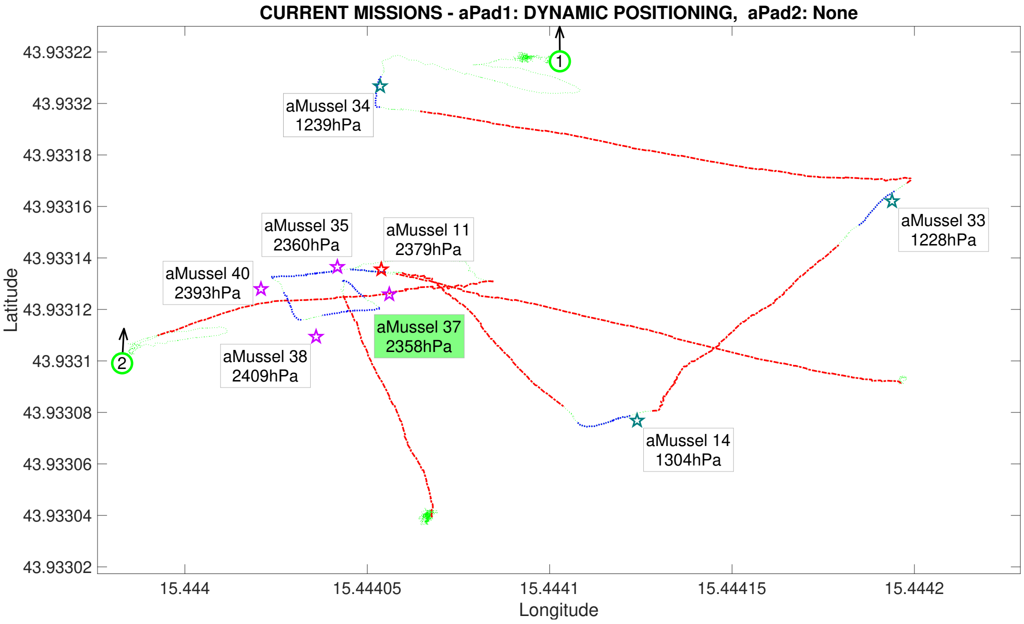


Figure 17. Outlier validation started in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

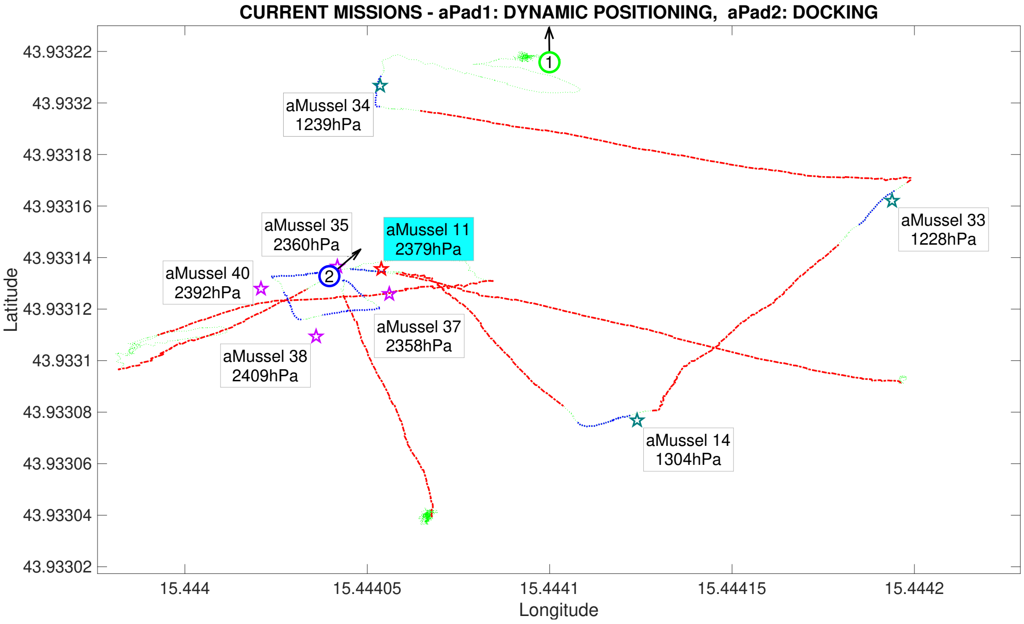


Figure 18. Outlier validation completed in the Anoxia scenario experiment results from the deployment phase, reconstructed and replayed from recorded vehicle data—Biograd na Moru testbed.

Figure 19 shows the pressure measurements each aMussel broadcast via the acoustic channel during the Biograd na Moru experiment described above. The previously mentioned offsets of the outlier aMussel and the verification group $\mathbf{M}_{\mathbf{vl}}$ can be clearly seen in the exchanged measurements. The data plot is divided into sections to show the system transitioning between the various phases of the algorithm.

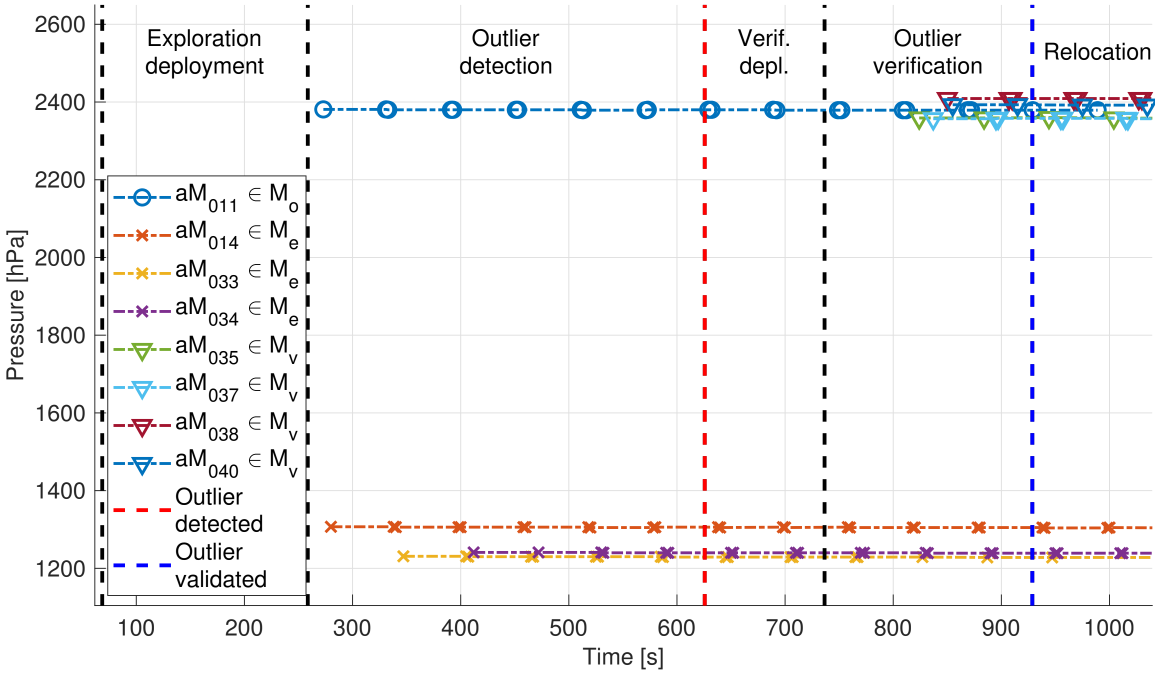
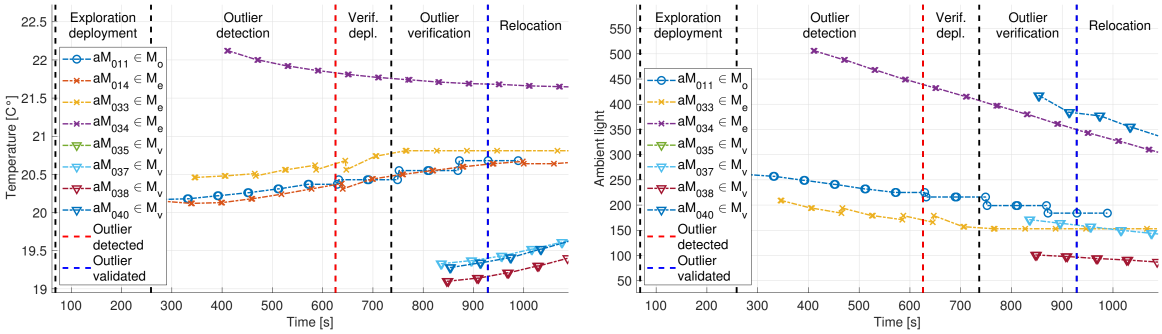


Figure 19. Pressure data exchanged via acoustic communication during the Biograd na Moru experiment. aMussel 11 is the outlier and was successfully verified as such.

In addition to pressure measurements, temperature and ambient light measurements were also exchanged between aMussels during the experiment, and are depicted in Figure 20 (note the consistency in data timestamps). A discrepancy in temperature data between the exploration (aM011, aM014, aM033, aM034) and verification group (aM035, aM037, aM038, aM040) is clearly apparent when looking at Figure 20a. This discrepancy is due to colder outside air temperature and the time constant of the sensor because the verification group stayed docked longer while waiting for the deployment phase to begin. However, as the agents sank to the bottom of the sea pool, the temperature started rising to the real temperature of the sea water. Additionally, aMussel 34 (purple) shows a slightly higher temperature than the other robots, possibly implying the existence of a warmer water current in the test bed.



(**a**) Temperature data (**b**) Ambient light data

Figure 20. Additional sensor data exchanged via acoustic communication during the Biograd na Moru experiment.

During the "Exploration deployment" phase, there was no pressure data exchanged due to the fact that all of the aMussels were still docked on the aPads or, later, drifting on the surface and thus their acoustic modems were above water. As they sank one by one and began exchanging acoustic messages (each message is represented by a marker in the data plot), measurements started showing on the acoustic channel. The red dashed line indicates the moment that the outlier aMussel was found.

After aMussel group M_{vl} was deployed, the "Outlier verification" phase began, in which the newly deployed group started to broadcast its own pressure measurements from the vicinity of the outlier aMussel. After they verified that the outlier aMussel wasn't faulty since their pressure measurements agreed, confirming the outlier's location as anomalous, the outlier aMussel broadcast its relocation message. This moment is indicated by a blue dashed line. The outlier's acoustic messages stop after that point, as it surfaces for pickup (its final message was broadcast just before the 1000 s mark in the plot).

Figure 21 depicts pressure measurements exchanged during one of the instances of the experiments performed in the Lagoon of Venice. The same paradigm was tested as in the instances described above, with the designated outlier's sensor measurements being "correct" (i.e., there was an anomaly to be discovered) and having the aMussels focusing on pressure sensor measurements. This testbed also represented a more challenging environment for the swarm—note, for example, that there is a longer delay before each of the aMussels start sending acoustic data, due to the fact that the aPads needed to cover greater distances over open water during the deployment phase and thus took longer to deploy them.

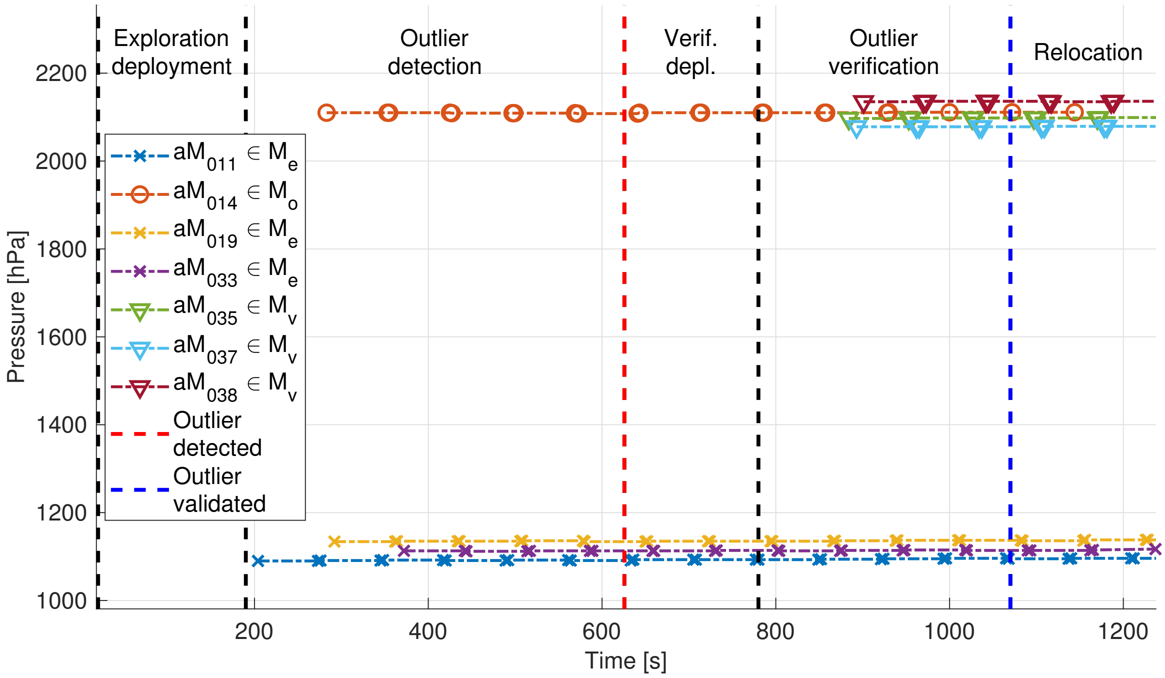
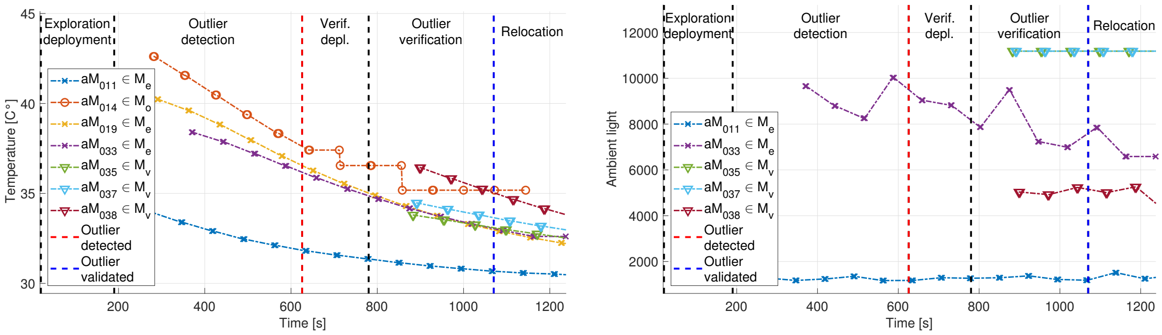


Figure 21. Pressure data exchanged via acoustic communication during the Venice experiment. aMussel 14 is the outlier and was successfully verified as such.

As in the Biograd na Moru experiment, Figure 22 shows the other broadcast measurements with the same consistency in data timestamps. The temperature data are displayed in Figure 22a. Before the start of the experiment, the aMussels were docked on the aPads under direct sunlight, which explains temperatures above 40 °C. As time progresses, measured temperatures drop to normal sea temperatures in the Venice Lagoon. Ambient light data shown in Figure 22b do not contain measurements from aMussels 14 and 19. This is due to the fact that these particular aMussels were not equipped with an ambient light sensor.



(**a**) Temperature data (**b**) Ambient light data

Figure 22. Additional sensor data exchanged via acoustic communication during the Venice experiment.

The aMussel exploration group $\mathbf{M_e}$ consisted of aMussels 11, 14, 19, and 33, with aMussel 14 being the designated outlier, while the verification group $\mathbf{M_{vl}}$ contained aMussels 35, 37, 38, and 39. This particular recorded dataset was chosen due to the fact that aMussel 39 from the verification group suffered a failure of its buoyancy system and did not sink to the bottom when it was supposed to, and thus did not participate in the exchange of acoustic messages for outlier verification. Nevertheless, the experiment was carried out successfully, and an environmental anomaly was determined to exist, with the outlier aMussel 14 surfacing for collection and relocation.

The step-by-step analysis of the data gathered during the proof-of-concept experiments serves to validate the functioning of the proposed distributed monitoring and fault/anomaly detection approach and demonstrate its implementation on real hardware, as well as its viability in a realistic marine environment. While the numerous functionalities and behaviours of the subCULTron swarm were tested previously (as described in [31]), this “anoxia scenario” served as a test of the swarm as a whole, relying on every aspect of it fulfilling its role, established robust communication protocols, and fault-tolerant behaviours.

The algorithm proposed and tested here makes use of the particular strengths of a distributed system in order to achieve autonomous long-term underwater monitoring, while engaging in dynamic topology reconfiguration and a type of self-maintenance by detecting and removing faulty agents from the active group. Combined with exploration strategies, the full monitoring loop could autonomously cover a significant area, creating a map of sensor values and pinpointing anomalous readings for the chosen value of interest.

4. Conclusions

This paper presented a novel paradigm for autonomous monitoring of a marine environment using a heterogeneous robotic swarm, as well as the field tests and experimental results achieved.

While the outlier detection algorithm used in this paper is effective in its simplicity and proved appropriate for the proof-of-concept experiments, one of the goals for continued development is testing a variety of more complex consensus algorithms, such as trust-based approaches. Another avenue of further research with the subCULTron swarm is incorporating different exploration strategies into the monitoring paradigm described here.

The main point of future work, while logistically challenging, is conducting long-term experiments while deploying the swarm at full scale. This will provide both robustness testing and gathering valuable data, as well as relevant system benchmarks helpful in identifying the main directions for potential development and system improvement. Combining the approach presented in this paper with the energy exchange and long-term autonomy related algorithms described in [32] will make it possible to use the developed swarm to its full environmental monitoring potential.

Author Contributions: Conceptualization, T.P., S.B. and N.M.; Formal analysis, A.B., I.L., B.A. and G.V.; Methodology, A.B., I.L., B.A. and G.V.; Software, A.B., I.L., B.A., G.V. and T.P.; Supervision, T.P., S.B. and

N.M.; Validation, A.B., I.L., B.A. and G.V.; Writing—original draft, A.B., I.L., B.A., G.V. and T.P.; Writing—review and editing, A.B., I.L., B.A., G.V., T.P., S.B. and N.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Croatian Science Foundation under Project IP-2016-06-2082 "Cooperative robotics in marine monitoring and exploration", and in part by the European Union through the ERDF project KK.01.1.1.01.0009 "DATACROSS - Advanced methods and technologies in Data Science and Cooperative Systems" and the EU H2020 FET-Proactive project "subCULTron", No. 640967. The work of doctoral students Barbara Arbanas and Ivan Lončar has been supported in part by the "Young researchers' career development project—training of doctoral students" of the Croatian Science Foundation funded by the European Union from the European Social Fund.

Acknowledgments: The authors wish to thank the members of the consortium of the project 'subCULTron' consisting of Artificial Life Lab—University of Graz, Unit of Social Ecology—Université Libre de Bruxelles, Cybertronica Research, IRCCyN—Ecole des Mines de Nantes, BioRobotics Institute—Scuola Superiore Sant'Anna, CORILA: Consortium for coordination of research activities concerning the Venice lagoon system and ISMAR Istituto di Scienze Marine.

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

Abbreviations

The following abbreviations are used in this manuscript:

API	Application Programming Interface
ASV	Autonomous Surface Vehicle
AUV	Autonomous Underwater Vehicle
GPS	Global Positioning System
GSM	Global System for Mobile Communications
IMU	Inertial Measurement Unit
IoUT	Internet of Underwater Things
UASN	Underwater Acoustic Sensor Network

References

1. Tagliapietra, D.; Aloui Bejaoui, N.; Bellafiore, D.; De Wit, R.; Ferrarin, C.; Gamito, S.; Lasserre, P.; Magni, P.; Mistri, M.; Pérez Ruzafa, A.; et al. The ecological implications of climate change on the Lagoon of Venice. *Unesco Digit. Libr.* **2011**, *4*, 1–48.
2. Ferrighi, A. Flooding and environmental challenges for Venice and its lagoon: State of knowledge. *Environmentalist* **2006**, *26*, 331.
3. Sorokin, Y.I.; Sorokin, P.Y.; Giovanardi, O.; Dalla Venezia, L. Study of the ecosystem of the lagoon of Venice, with emphasis on anthropogenic impact. *Mar. Ecol. Prog. Ser.* **1996**, *141*, 247–261. [CrossRef]
4. Argese, E.; Cogoni, G.; Zaggia, L.; Zonta, R.; Pini, R. Study on redox state and grain size of sediments in a mud flat of the Venice Lagoon. *Environ. Geol. Water Sci.* **1992**, *20*, 35–42. [CrossRef]
5. Solidoro, C.; Pecenik, G.; Pastres, R.; Franco, D.; Dejak, C. Modelling macroalgae (Ulva rigida) in the Venice lagoon: Model structure identification and first parameters estimation. *Ecol. Model.* **1997**, *94*, 191–206. [CrossRef]
6. Lang, F.; von der Lippe, M.; Schimpel, S.; Scozzafava-Jaeger, T.; Straub, W. Topsoil morphology indicates bio-effective redox conditions in Venice salt marshes. *Estuar. Coast. Shelf Sci.* **2010**, *87*, 11–20. [CrossRef]
7. Akyildiz, I.F.; Pompili, D.; Melodia, T. Underwater acoustic sensor networks: Research challenges. *Hoc Netw.* **2005**, *3*, 257–279. [CrossRef]
8. Tuna, G.; Gungor, V.C. A survey on deployment techniques, localization algorithms, and research challenges for underwater acoustic sensor networks. *Int. J. Commun. Syst.* **2017**, *30*, e3350. [CrossRef]
9. Kao, C.C.; Lin, Y.S.; Wu, G.D.; Huang, C.J. A Comprehensive Study on the Internet of Underwater Things: Applications, Challenges, and Channel Models. *Sensors* **2017**, *17*, 1477. [CrossRef]
10. Shkurti, F.; Xu, A.; Meghjani, M.; Higuera, J.C.G.; Girdhar, Y.; Giguere, P.; Dey, B.B.; Li, J.; Kalmbach, A.; Prahacs, C.; et al. Multi-domain monitoring of marine environments using a heterogeneous robot team. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012; pp. 1747–1753.

11. Girdhar, Y.; Xu, A.; Shkurti, F.; Gamboa Higuera, J.C.; Meghjani, M.; Rekleitis, I.; Dudek, G. Monitoring Marine Environments using a Team of Heterogeneous Robots. In Proceedings of the RSS 2012 Workshop on Robotics for Environmental Monitoring (WREM 2012), Vilamoura, Portugal, 7–12 October 2012.
12. Jaffe, J.S.; Franks, P.J.; Roberts, P.L.; Mirza, D.; Schurgers, C.; Kastner, R.; Boch, A. A swarm of autonomous miniature underwater robot drifters for exploring submesoscale ocean dynamics. *Nat. Commun.* **2017**, *8*, 1–8. [CrossRef]
13. Molchanov, A.; Breitenmoser, A.; Sukhatme, G.S. Active drifters: Towards a practical multi-robot system for ocean monitoring. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 545–552.
14. Zhang, B.; Sukhatme, G.S.; Requicha, A.A. Adaptive sampling for marine microorganism monitoring. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566), Sendai, Japan, 28 September–2 October 2004; Volume 2, pp. 1115–1122.
15. Vasilescu, I.; Kotay, K.; Rus, D.; Dunbabin, M.; Corke, P. Data Collection, Storage, and Retrieval with an Underwater Sensor Network. In Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, SanDiego, CA, USA, 2–4 November 2005; pp. 154–165. [CrossRef]
16. Ribeiro, F.J.L.; de Castro Pinto Pedroza, A.; Costa, L.H.M.K. Underwater monitoring system for oil exploration using acoustic sensor networks. *Telecommun. Syst.* **2015**, *58*, 91–106. [CrossRef]
17. Burgard, W.; Moors, M.; Stachniss, C.; Schneider, F.E. Coordinated multi-robot exploration. *IEEE Trans. Robot.* **2005**, *21*, 376–386. [CrossRef]
18. Varughese, J.C.; Thenius, R.; Leitgeb, P.; Wotawa, F.; Schmickl, T. A Model for Bio-Inspired Underwater Swarm Robotic Exploration. *IFAC-PapersOnLine* **2018**, *51*, 385–390. [CrossRef]
19. Birk, A. Distributed Communicative Exploration under underwater communication constraints. In Proceedings of the 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, Japan, 1–5 November 2011; pp. 339–344.
20. Fossum, T.O.; Eidsvik, J.; Ellingsen, I.; Alver, M.O.; Fragoso, G.M.; Johnsen, G.; Mendes, R.; Ludvigsen, M.; Rajan, K. Information-driven robotic sampling in the coastal ocean. *J. Field Robot.* **2018**, *35*, 1101–1121. [CrossRef]
21. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection. *ACM Comput. Surv.* **2009**, *41*, 1–58. [CrossRef]
22. Mazdin, P.; Arbanas, B.; Haus, T.; Bogdan, S.; Petrovic, T.; Miskovic, N. Trust Consensus Protocol for Heterogeneous Underwater Robotic Systems. *IFAC-PapersOnLine* **2016**, *49*, 341–346. [CrossRef]
23. Franceschelli, M.; Giua, A.; Pisano, A. Finite-Time Consensus on the Median Value With Robustness Properties. *IEEE Trans. Autom. Control* **2017**, *62*, 1652–1667. [CrossRef]
24. Vasiljevic, G.; Petrovic, T.; Arbanas, B.; Bogdan, S. Dynamic Median Consensus for Marine Multi-Robot Systems Using Acoustic Communication. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5299–5306. [CrossRef]
25. Saldaña, D.; Assunção, R.; Campos, M.F.M. A distributed multi-robot approach for the detection and tracking of multiple dynamic anomalies. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 1262–1267.
26. Luo, T.; Nagarajan, S.G. Distributed Anomaly Detection Using Autoencoder Neural Networks in WSN for IoT. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
27. Mahmoud, M.S.; Khalid, H.M. Distributed Kalman filtering: A bibliographic review. *IET Control Theory Appl.* **2013**, *7*, 483–501. [CrossRef]
28. Olfati-Saber, R. Distributed Kalman filtering for sensor networks. In Proceedings of the 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007. [CrossRef]
29. Caliskan, F.; Hajiyev, C. Aircraft sensor fault diagnosis based on Kalman filter innovation sequence. In Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No.98CH36171), Tampa, FL, USA, 18 December 1998. [CrossRef]
30. Thenius, R.; Moser, D.; Varughese, J.C.; Kernbach, S.; Kuksin, I.; Kernbach, O.; Kuksina, E.; Mišković, N.; Bogdan, S.; Petrović, T.; et al. subCULTron—Cultural Development as a Tool in Underwater Robotics. In *Artificial Life and Intelligent Agents*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; pp. 27–41._3. [CrossRef]
31. Lončar, I.; Babić, A.; Arbanas, B.; Vasiljević, G.; Petrović, T.; Bogdan, S.; Mišković, N. A Heterogeneous Robotic Swarm for Long-Term Monitoring of Marine Environments. *Appl. Sci.* **2019**, *9*, 1388. [CrossRef]

32. Babić, A.; Vasiljević, G.; Mišković, N. Vehicle-in-the-Loop Framework for Testing Long-Term Autonomy in a Heterogeneous Marine Robot Swarm. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4439–4446. [CrossRef]
33. Barcellona, S.; Brenna, M.; Foiadelli, F.; Longo, M.; Piegari, L. Analysis of ageing effect on Li-polymer batteries. *Sci. World J.* **2015**, *2015*, 979321. [CrossRef] [PubMed]
34. Babić, A.; Mandić, F.; Vasiljević, G.; Mišković, N. Autonomous docking and energy sharing between two types of robotic agents. *IFAC-PapersOnLine* **2018**, *51*, 406–411. [CrossRef]