drones

Article

DEDG: Cluster-Based Delay and Energy-Aware Data Gathering in 3D-UWSN with Optimal Movement of Multi-AUV

Reem Alkanhel [1,*], Amir Chaaf [2], Nagwan Abdel Samee [1], Manal Abdullah Alohali [3], Mohammed Saleh Ali Muthanna [4], Dmitry Poluektov [5] and Ammar Muthanna [5,6]

[1] Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[2] School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[3] Information Systems Department, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11564, Saudi Arabia
[4] Institute of Computer Technologies and Information Security, Southern Federal University, 347922 Taganrog, Russia
[5] Department of Applied Probability and Informatics, Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya, 117198 Moscow, Russia
[6] Department of Telecommunication Networks and Data Transmission, The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, 193232 Saint Petersburg, Russia
* Correspondence: rialkanhal@pnu.edu.sa

check for updates

Citation: Alkanhel, R.; Chaaf, A.; Samee, N.A.; Alohali, M.A.; Muthanna, M.S.A.; Poluektov, D.; Muthanna, A. DEDG: Cluster-Based Delay and Energy-Aware Data Gathering in 3D-UWSN with Optimal Movement of Multi-AUV. *Drones* **2022**, *6*, 283. https://doi.org/10.3390/drones6100283

Academic Editors: Saeed Hamood Alsamhi, Faris A. Almalki, Jahan Hassan, Sudheesh Puthenveettil Gopi, Alexey V. Shvetsov and Deepak GC

Received: 30 August 2022
Accepted: 26 September 2022
Published: 1 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 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 (https://creativecommons.org/licenses/by/4.0/).

Abstract: The monitoring of underwater aquatic habitats and pipeline leakages and disaster prevention are assisted by the construction of an underwater wireless sensor network (UWSN). The deployment of underwater sensors consumes energy and causes delay when transferring the gathered sensed data via multiple hops. The consumption of energy and delays are minimized by means of an autonomous unmanned vehicle (AUV). This work addresses the idea of reducing energy and delay by incorporating an AUVs-assisted, three-dimensional UWSN (3D-UWSN) called DEDG 3D-UWSN. Energy in the sensor nodes is saved by clustering and scheduling; on the other hand, the delay is minimized by the movement of the AUV and inter-cluster routing. In clustering, multi-objective spotted hyena optimization (MO-SHO) is applied for the selection of the best sensor for the cluster head, which is responsible for assigning sleep schedules for members. According to the total number of members, an equal half of the members is provided with sleep slots based on the energy and hop counts. The redundancy in the gathered data is eliminated by measuring the Hassanat distance. Then, the moving AUV is able to predict its movement by the di-factor actor–critic path prediction method. The mid-point among the four heads is determined so that the AUV can collect data from four heads at a time. In cases where the waiting time of the CH is exceeded, three-step, inter-cluster routing is executed. The three steps are the discovery of possible routes, ignoring the longest paths and validating the filtered path with a fuzzy–LeNet method. In this 3D-UWSN, the sensed data are not always normal, and, hence, a weighted method is presented to transfer emergency events by selecting forwarders. This work is implemented on Network Simulator version 3.26 to test the results. It achieves better efficiency in terms of data collection delay, end-to-end delay, AUV tour length, network lifetime, number of alive nodes and energy consumption.

Keywords: autonomous unmanned vehicle; data gathering; emergency event; inter-cluster routing; scheduling; underwater WSN

1. Introduction

An underwater wireless sensor network (UWSN) is an underwater environment containing oceanographic sensors and vehicles. These sensors are low-powered devices that are capable of sensing surroundings up to a particular distance, and they transmit the sensed data using acoustic signals. The sensor data are collected by the sink node, which is located over the on-shore or above the water level, for further analysis. In this way, UWSNs are supported for a variety of real-world applications, such as in aquatic habitat/health surveillance, underwater pollution level estimation, fishing and others [1–4]. The sensed data are transmitted to the other end sink for analysis. Here, the sensor nodes are equipped with a limited amount of battery, which has to be saved in order to extend its lifetime. Clustering is an efficient solution for saving energy as well as for gathering the sensed data [5–7]. After creating clusters, a cluster head (CH) is selected, which is responsible for collecting data from its members. The CH is selected by evaluating any one of its potentialities, energy level, distance, node degree, moving speed, position, signal strength and others. These computations are formulated into fitness function using a bio-inspired optimization algorithm. Swarm optimization algorithms are involved in the process of fitness estimation [8–11].

A UWSN becomes more flexible with the deployment of a moving autonomous unmanned vehicle (AUV). The AUV either collects data from individual sensors or from the CH. An AUV-aided UWSN mitigates energy consumption at the sensors by eliminating transmission of sensed data that use longer paths with n-hops [12]. The use of a AUV collects data directly from the sensors; this makes it possible to avoid the transfer of data through a chain of several sensors in a succession. Decreasing the number of transmissions efficiently balances the overall energy consumption and length of delay. Location prediction and path planning of the AUV are challenging since they impact on network lifetime. The route towards the AUV is selected for data transmission [13–17]. The trajectory of the AUV is fixed in some cases in which the moving path is pre-defined by the sink. This trajectory is also made dynamic by planning the movement of the AUV for data collection. The routing in a UWSN is majorly concentrated on the minimization of the energy and delay [18–20]. Network lifetime is improved by efficient path planning with evolutionary algorithms [21,22].

Even though an AUV is appointed in a UWSN for data gathering, it is essential to plan the path properly. If a single AUV is used, it is easier to carry out path planning, but the coverage of the AUV over the entire network area still consumes more energy. So, more than one AUV is employed for performing data gathering with the aim of improving network lifetime. The challenges in an AUV-aided UWSN are illustrated below:

- Energy saving in underwater sensors is challenging because the nodes are equipped with an amount of battery within which it can perform sensing and transmission;
- Delay in data transmission increases due to the involvement of multiple intermediate hops and waiting time. Transmitting the data through multiple sensors in a relay pattern, and also the sensor waiting time before the AUV’s arrival for data collection, causes high delays;
- Dynamic topology in the network due to the movement of sensors in relation to the ocean current changes over time.

In this paper, a 3D-UWSN architecture was designed with the objective of minimizing energy consumption and transmission delay. In order to achieve this objective, the processes of clustering, scheduling, AUV movement and inter-cluster routing were designed. The energy was saved by optimal CH selection and assigning sleep slots. Then, the data gathering delay was reduced by appropriate movement of AUV and inter-cluster routing. The architecture was composed of two AUVs to serve two depths of sensor node.

1.1. Motivation

The motivation of this paper was overcoming the challenges in UWSNs, which was also the key objective of this research. In a two-dimensional UWSN (2D-UWSN), sensor nodes are able to be deployed in the spatial location, whereas, in an ocean environment, it is essential to handle the monitoring at different depths. For this purpose, a 3D-UWSN is launched in which underwater sensors are employed at different depths [23,24]. The challenging issues created by 3D-UWSNs are the consumption of higher energy and the delay on sensor nodes due to data transmission. The gathering of sensed data is the key task from which the environmental changes are analyzed. As a solution, a cluster is constructed and then inter-cluster routing is performed with or without the participation of an AUV [25,26]. An AUV moves in and around the network to gather data from sensors. When using a single AUV, it takes quite a long waiting time for sensors to transmit their sensed data, and so multiple AUVs are introduced [27]. The utilization of AUVs enables efficient collection of data from specifically concerned sensor nodes and minimizes energy consumption and delay for the following reasons:

- The collected sensor information is not transmitted via multiple hops; it is directly sent to the AUV when it arrives near to the sensor. Therefore, the direct transmission of sensed data also reduces the energy consumption of the sensors and the transmission delays;
- Data gathering from CHs also reduces the overall energy consumption of the sensors and the gathering delay, since it is not required for all the sensors to forward their data to the AUV or to wait in order to deliver the data;
- Planning of multiple AUV paths appropriately minimizes the tour length, which reduces the delay in transmitting the data from sensors to sink;
- Allotment of sleep schedules for sensors enables the reduction of a considerable amount of sensing energy.

The design of the 3D-UWSN was motivated towards the reduction of delay and improvement of network lifetime with the incorporation of clustering, data gathering and supportability of delay-aware event message transmission. Events are instances when emergency information occurs; such information is more sensitive, and, hence, it needs to be transmitted as early as possible to avoid heavier damages. Moving on with this motivation, the major, problematic issues in this research were identified and resolved with reinforcement learning, optimization algorithms and artificial intelligence methods. This enabled us to achieve the objectives of this proposed research work, and the achievement is discussed by comparing the designed system with previous AUV-assisted UWSN designs.

1.2. Contribution of the Paper

The major contributions of this paper are defined with respect to the key objectives of improving network lifetime and reducing delay in a 3D-UWSN with the assistance of AUVs. The contributions are the following:

- Clustering and optimal CH selection with multi-objective spotted hyena optimization (MO-SHO), which formulates fitness using a sensor's lifetime, degree and centrality. The elected CH takes responsibility for allotting sleep slots and redundancy eliminations on the sensor nodes by taking into consideration some parameters, such as sensor's energy and hop counts, to make an efficient decision on whether the sensor status is to be executed while sleep, awake or idle, and the Hassanat distance metric is used for similarity data measurement for redundancy elimination;
- The data gathering AUVs predict their position dynamically by executing di-factor actor–critic path (DACP) prediction. The mid-point-oriented central position is determined to collect data from four CHs. The ability of the CH is identified by evaluating buffer, data collection delay, received signal strength indicator (RSSI) and data size in DACP prediction;
- A three-step (TS), inter-cluster routing method is executed only when the waiting time for the AUV is exceeded. The TS method firstly discovers the route, secondly, filters long routes and, lastly, validates the route. The fuzzy–LeNet method for route validation is used for estimating energy consumption, fairness, synthesis speed and efficiency;
- Emergency event data are transmitted immediately without any delay by means of selecting a forwarder with heavier weights estimated by distance, load and residual energy.

1.3. Outline of the Paper

This paper is organized into the following sections: Section 2 illustrates previous research works on AUV-assisted UWSNs that minimize delay and energy consumption; Section 3 keenly highlights the problem and the idea to overcome the identified, problematic issue; Section 4 elaborates the proposed solutions and algorithm descriptions; Section 5 demonstrates the tested environment with comparative evaluation; and Section 6 gives the conclusion of this research and extends it in future directions.

2. Literature Review

In this section, previous research papers are studied, and their limitations and demerits are highlighted. The sensed environmental data are transmitted to the other end sink. For this purpose, routing is performed to transmit data via intermediate sensors. Then, the involvement of the AUV for data gathering is also detailed in this section.

2.1. Clustering and Data Forwarding

A data forwarding procedure using a Q-learning model was proposed [28]. Initially, a beacon message was broadcast using a sink through which the sensor identifies its neighbors. The neighbor node was updated with RSSI and angle of arrival. If the sink node was not present within the sensor's one-hop distance, then a forwarder was selected using the Q-learning process according to the transmission probability. The increase in the number of forwarders drained energy in the source sensor as well as in the forwarder. In [29], a UWSN was modeled as disjoint concentric coronas in six partitions. Here, a source sensor node selected intermediate sensors from every wedge and then, using the path, their sensed data reached the sink node. However, this work balanced energy consumption with the formation of a corona for a large-scale environment, which was tedious.

A depth-based routing mechanism named the energy-balanced, efficient and reliable routing (EBER2) protocol was developed in [30]. A forwarder was selected using three constraints: weighted depth difference, potential forwarder node (PFN) and residual energy. The data transmitting nodes determined distance and selected the nearest sink to deliver the data. The sensor nodes nearer to the sink drained more energy due to the transmission of the sensed data from the nodes in the network. A self-organizing and scalable routing protocol (SOSRP) was operated with the computation of hop count and distance [31]. In this protocol, the neighbors were discovered by exchanging messages and then hop count; distance was estimated for path prediction. The path with the shortest hop count was used for data transmission. In this work, the selected nodes performed poorly in relation to energy consumption, and, hence, the stability of the path was not hardy.

An energy-efficient multipath routing (E^2MR) was developed in this paper, which selected a forwarding node from the constructed priority table [32]. By exchanging control packets, the priority table was constructed with the entities of residual energy, distance and priority value. The priority value was computed based on the energy and the depth of the sensor. Here, the node with higher priority value was selected as a forwarding node. Routing was performed effectively with the construction of a cluster that eventually minimized energy consumption. A multi-layer, cluster-based, energy-efficient (MLCEE) protocol was proposed, in which the CH was selected by computing the Bayesian probability and residual energy of the particular node [33]. After the construction of the clusters, the members were allotted with time division multiple access (TDMA) time slots [34]. The CH selected a forwarding node by determining residual energy, hop and probability as a fitness value.

A QoS-aware, evolutionary, cluster-based routing protocol (QERP) was presented in a 3D environment [35]. The operators involved in the genetic algorithm (GA) were crossover and mutation operators. The key constraints that were taken into account were identity, energy and RSSI. This GA was time consuming, and, also, it was tedious when multiple sensor nodes participated. An adaptive node clustering technique was developed for a smart ocean UWSN known as SOSNET [36]. The moth–flame optimization (MFO) algorithm was applied for optimal performance of routing. The CH was selected by estimating transmission range, residual energy, node density and load balance factor. The fitness value for the optimization algorithm was computed using residual energy, distance with the neighbors, load balancing factor and weight value. The MFO algorithm was slow in convergence, which failed in routing. Therefore, the routing of sensed data packets was required to be faster and effective with the reduction of delay and energy consumption of sensors. Delay in the selection of route also increased the delay in data transmission.

2.2. Data Gathering by AUV Path Planning

As mentioned above, the UWSN environment is involved in a variety of applications. Pipeline monitoring was addressed with a UWSN using an AUV [37]. The sensed data were collected by AUV and then forwarded to the surface sink. The AUV followed a linear sensor network segment model to gather the data sequentially. If the sensed data were emergency data, then the AUV skipped the entire intermediate sensor and reached the event-sensed sensor to gather its data. Therefore, the emergency data were provided with higher priority; however, the use of a single AUV was not sufficient to support a large-scale network, and also it increased waiting time for delivery of data. Data gathering was presented with an evolutionary algorithm for the path planning of an AUV [38]. The PSO was used in estimating the optimal waypoint at which the data have to be collected. This PSO algorithm was slow in convergence, which took more time in optimal selection.

A hybrid optimization algorithm was designed for predicting an optimal solution for AUV tours [39]. The work proposed an integrated, quantum-behaved PSO with improved ant colony optimization (ACO) to predict a shorter tour path. Initially, the lowest number of waypoints was generated, and the points to be visited were selected using the algorithm. The shortest path was chosen using ACO algorithm, which extended the network lifetime by reducing the trajectory length. The conventional problem in this ACO is that it is slower in convergence. Heuristic, adaptive sink sensor set selection (HAS^4) was designed to enrich network lifetime [40]. This HAS^4 was developed for centralized and distributed environments in which the sensor lifetime plays a key role. Sensor lifetime was the only metric computed and sorted among all the sensor nodes and then a set of sensors was selected, which the AUV visited to collect data. The AUV tour selected sensor nodes by minimum lifetime, which tended to increase the tour length.

A swarm hyper-heuristic algorithm (SHH) was associated with local path planning and global path planning [41]. The global path was identified using an A* algorithm, and the local path prediction was subsequently performed using time, obstacles and AUV motion. In general, this A* algorithm involves multiple computations that increase the complexity of planning the path of AUVs. A prediction-based delay optimization data collection algorithm (PDO-DC) was presented to mitigate the collection delay [42]. The clusters were created and then they were updated; further, an AUV path was predicted. According to this work, the data collected at CH had to request that the AUV collect the gathered data. Data collection performed in this way certainly increased the traveling length, and also more than one CH could request at a time. By analyzing the existing works, the major demerits in UWSNs used for routing and data gathering were identified and resolved with particular solutions.

3. Problem Description

To date, 3D-UWSN architecture has largely been designed for efficient data gathering and routing with the aim of extending network lifetime by saving energy at deployed sensor nodes. Still, these contributions resulted in a few problematic issues, which are discussed in this section. In a UWSN, the AUV plays a significant role in collecting data from the network. Energy saving is modeled with clustering and sleep–wakeup scheduling called the AUV-assisted, energy-efficient clustering (AEC) mechanism [43]. The AUV selects the CH on its first traversal and collects data during its second traversal. The awake time for nodes is based on the sensing interval, guard delay, transmission of data packet time and transmission of acknowledgment packet time. CH selection is poor due to the estimation only including energy. Here, the time to wakeup of sensors is predicted from the delay and transmission time, which are the key parameters used to forward the data, since it is essential to measure energy for allotting sleep slots.

Clustering-based data collection using an AUV is designed to balance the energy consumption [44,45]. In a location prediction-based data collection scheme (ALP), the AUV selects to visit a node with a smaller collection delay. According to the collection delay, the trajectory path of the AUV is constructed. Then, a bipartite K-means-based clustered network uses an AUV for data collection and prefers data collection based on the residual energy and distance of the node. The target nodes are selected one after the other, i.e., the AUV has the ability to collect data from multiple nodes, but, here, the data from CHs are collected one at a time, which increases the delay in data delivery. Increase in the trajectory length increases the data collection delay; also, the deployment of a single AUV increases tour length.

Two AUVs are presented for data gathering with an enhanced lawn mower pattern path (ELMPP) [46]. The path planning follows straight line segments and turning line segments. Further, the relay node is established by a communication link with the prediction of the RSSI value. The same sensor is selected as a relay, which tends to drain more energy at the relay nodes, and the RSSI is capable of determining the strength of the communication link but not the strength of the node. The common problems existing in prior research works are:

- In a UWSN, using a single AUV for data collection cannot collect data in a large-scale environment, which increases delay, and also it fails to deliver emergency data promptly. Pre-defined path planning of a single AUV causes a longer waiting time for the nodes before the AUV reaches a nearer position;
- In a UWSN, continuous sensing of the environment consumes more energy, and transmission in multiple hops towards the sink also consumes more energy across all the nodes in the network, which reduces network lifetime.

The above-stated problems were overcome in the proposed research by the design of cluster-based delay and energy-aware data gathering in a 3D-UWSN.

4. Proposed 3D-UWSN Design

The 3D-UWSN is a network model designed for efficient data gathering and immediate transmission of event messages. Events are the emergency accidents that happen in the underwater environment, and they are detected by the sensors. This kind of sensed information is sensitive, and it needs to be forwarded faster to allow for corresponding countermeasures. The proposed network design is operated through the processes of clustering, routing and data gathering by different algorithms. This section is composed of the network model, clustering, AUV movement for data gathering, inter-cluster routing and event message delivery.

4.1. Network Model

The proposed DEDG 3D-UWSN is constructed to monitor the underwater marine environment. A 3D-UWSN absorbs its surroundings by collecting data and transmitting the information. This 3D-UWSN is constructed as $x_i \times y_i \times z_i$, i.e., the coordinates (x_i, y_i, z_i) of the network area in 3D. This 3D structure is divided into two levels, L_1 and L_2, based on the depths of the ocean. L_1 and L_2 are assumed to cover shallow ocean and deep ocean, respectively. In this network, N and M numbers of sensor nodes are employed in L_1 and L_2, respectively. L_1 is composed of l number of sensors as $L_1 = \{s_1, s_2, s_3, \ldots, s_l\}$, and L_2 is composed of m number of sensors as $L_2 = \{s_1, s_2, s_3, \ldots, s_m\}$.

The deployed sensors are capable of performing underwater environment sensing tasks and transmitting the collected information to their neighboring sensors or the sink node through the deployed AUVs. The sensors are equipped with acoustic transceivers, namely, a vertical and horizontal transceiver to allow communication between them and the

AUV. The sensing model of the sensor nodes is a spherical ball with radius R_s. The sensing is performed within this radius. We assume that these sensors are sea-bottom anchored and are considered quasi-stationary due to the ocean current and energy constrained; each sensor node has its unique identifier, and the sensors are homogeneous in terms of initial energy and communication range. The location of the sensors is known. The AUV moves above level 1 and level 2 and collects data.

We assume that the AUVs start from their source places, dive to the CHs (each elected cluster head that gathers the data from its cluster members) and collect data from all the CHs successively, then arrive at the sink node. The AUVs act as temporary sinks. The AUVs are equipped with both an acoustic modem and a radio transmitter to communicate with the sensor nodes and the sink node, respectively. The AUVs' initial velocity is known, and the velocity is not constant during the AUVs' movement path; the energy of the AUVs is unlimited. A single sink node is equipped with both acoustic transceivers to handle multiple, parallel communications with the sensor nodes and also a long-range RF and satellite transmitters to communicate with the AUVs in order to receive the collected data and also to transmit data to the on-shore station. The data are gathering by the AUVs from the CHs; here, the two AUVs are appointed as AV_1 and AV_2, respectively. AV_1 collects data from L_1 CHs and AV_2 from L_2 CHs. A super node acts as gateway GW in between the L_1 and L_2, which is responsible for selecting the inter-cluster route when the waiting time in the CHs is exceeded. The designed DEDG 3D-UWSN architecture is depicted in Figure 1.

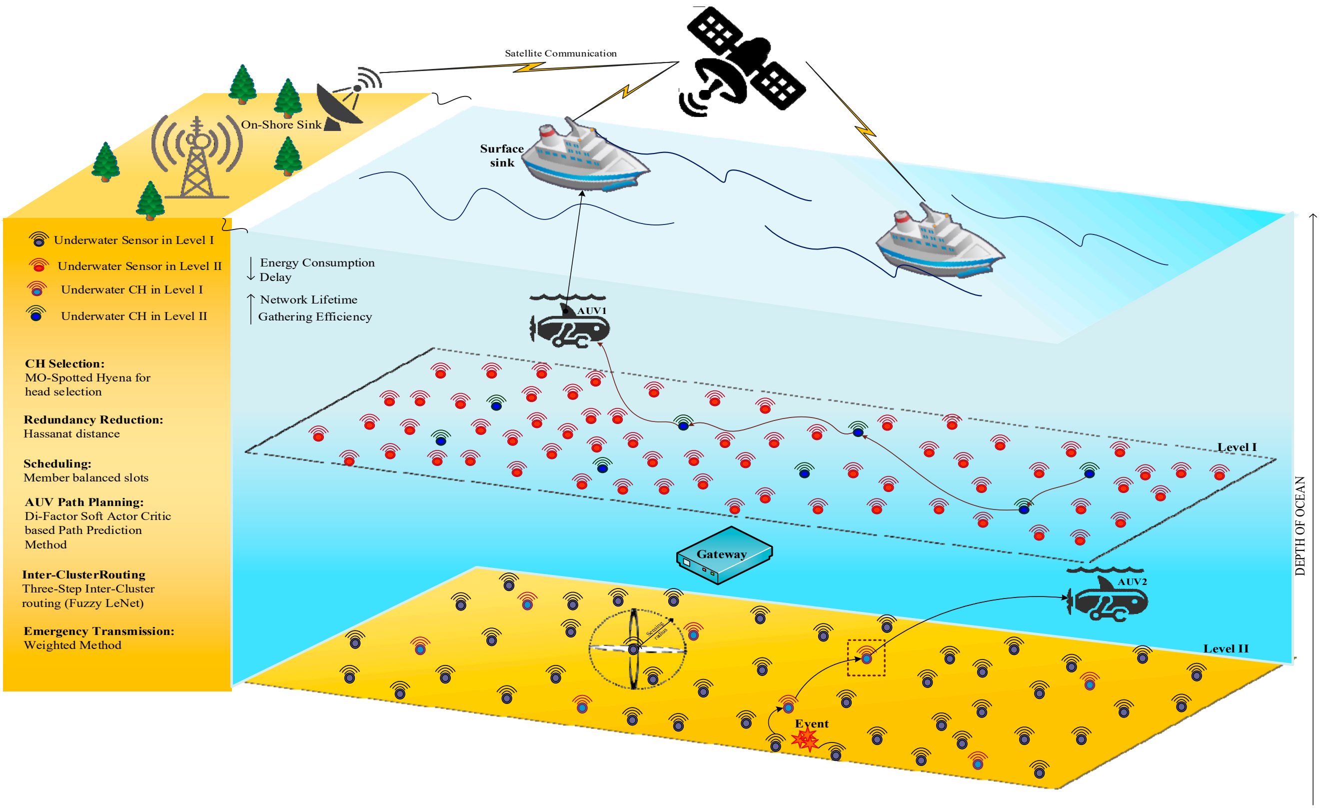

Figure 1. DEDG 3D-UWSN architecture.

This DEDG 3D-UWSN supports the processes of clustering, scheduling, movement prediction of AUVs, inter-cluster routing and event message delivery. The energy consumption in the underwater sensor depends on the performance of the following processes: sensing, transmitting, receiving and idle. The sensor nodes are capable of sending and receiving 1-bit data within the network. In the DEDG 3D-UWSN, the sensor nodes are clustered, and an optimal CH is selected. The CH is responsible for gathering sensed data from the active members in the cluster. The moving AUV collects data from four CHs at a time. This ensures the minimization of data gathering delay from the CHs at each level. The CHs are selected with the MO-SHO algorithm, which takes into account the node parameters. Every CH waits for a certain time period until the waiting time for the arrival of the AUV. The AUV moves above level 1 and level 2 and collects data. If the waiting time is exceeded, then the gateway helps in selecting an inter-cluster routing path towards the sink. CHs are appointed with a threshold waiting time in order to reduce the time delay that can be caused in cases where the AUV arrival time exceeds the waiting time. Another option is selected through the intervention of a gateway that helps to select an inter-cluster routing path to forward the data towards the AUV to the sink.

4.2. Clustering

In this proposed research DEDG 3D-UWSN, the sensor nodes are clustered into several clusters, and an optimal CH is selected in each cluster using an MO-SHO algorithm according to some key parameters. Initially, the CHs are selected, and the sensors within their coverage are added as cluster members (CMs). These sensors are affiliated to that CH and form a cluster. The selection of the CHs is made in relay; a CH is selected for a period of time until another sensor is selected as the next CH in turn. The selection of CHs is therefore made alternately. The cluster heads are responsible for managing the nodes within the cluster; therefore, the clustering with optimal head selection reduces the energy consumption in data gathering.

Multi-Objective Spotted Hyena Optimization (MO-SHO) Algorithm

Spotted hyenas are wild animals that usually live and hunt in groups and rely on a network of trusted friends with the ability to recognize the location of the prey. MO-SHO is a meta-heuristic algorithm modeled by analyzing the social hunting behavior of spotted hyenas. MO-SHO imitates the cohesive clusters between trusted spotted hyenas which are helpful for efficient co-operation and maximize the fitness. In this research, the hunting technique and the social relations of spotted hyenas are mathematically modeled to design the MO-SHO and perform optimization. To define the behavior of spotted hyenas mathematically, we suppose that the best search agents (the best hyenas hunters) are whichever have optimum knowledge of the location of the prey. The other search agents (other hyena followers) make a cluster towards the best search agent and save the best solutions obtained to update their positions.

The MO-SHO algorithm (Algorithm 1) is used for clustering with optimal head selection that reduces the energy consumption in data gathering.

The mathematical model of the MO-SHO behavior is formulated as [47]:

$$\vec{d}_H = \left| \vec{C}_1 \times \vec{P}_b(t) - \vec{P}_H(t) \right| \tag{1}$$

$$\vec{P}_H(t+1) = \vec{P}_T(t) - \vec{C}_2 \times \vec{d}_H \tag{2}$$

The distance between the hyena is $\vec{d}_H$ at t iteration. The terms $\vec{C}_1$ and $\vec{C}_2$ are the vector coefficients, $\vec{P}_H$ represents the current position of the hyena and the $\vec{P}_b$ vector defines the best position. The best positions of the hyenas are given as:

$$N_H = Sol\left(\vec{P}_{h_1}, \vec{P}_{h_1+1}, \vec{P}_{h_1+2}, \ldots, \left(\vec{P}_{h_1} + \vec{r}\right)\right) \tag{3}$$

N_H is the total number of hyenas, $\vec{r}$ indicates the random vector whose values are [0.5, 1] and their solutions are represented as sol, which is the best positions of the hyenas.

Hyenas are the search agents that are the sensors from which the best is chosen. The last-stored, best solution is given as:

$$\vec{P}_H(t+1) = \frac{\vec{G}_H}{N_H} \tag{4}$$

Here, $\vec{G}_H$ represents the group of optimal solutions that is obtained based on the position of the best hyenas. In fitness, the node degree is defined as the total number of neighbors connected with the particular node. The node degree N_d for the sensor in L_1 is mathematically formulated by:

$$N_d = \frac{\sum_1^l (s_I, s_j)}{(s_l - 1)} \tag{5}$$

s_l is the total number of sensors that exists in L_1, and I and j are the two sensor nodes which are 1 if the sensors are linked; otherwise they are 0. Then, the node lifetime is expressed as:

$$N_L = \frac{RE}{ED} \tag{6}$$

The node lifetime NL is computed from the residual energy of the sensor nodes RE to that of the energy drain rate ED. Along with N_d and N_L, the sensor's closeness centrality N_{cc} for the L_1 sensor is determined as:

$$N_{cc} = \left[\frac{\sum_1^l d(s_i, s_j)}{(s_l - 1)}\right]^{-1} \tag{7}$$

The distance between sensors is represented as $d(s_i, s_j)$; this closeness centrality defines the closeness of the particular sensor node to their neighboring nodes. The hyenas are the sensor nodes whose individual fitness is estimated, and their position is updated accordingly. The fitness for level-1 sensor nodes is given as:

$$F(L_1) = \{f_1,\ f_2, \ldots f_l\} \tag{8}$$

$$f_1 = \left(N_{d(1)},\ N_{L(1)}, N_{cc(1)}\right) \tag{9}$$

The fitness-based determination for each sensor is significant for selecting the best one within the search space. A node with higher stability, i.e., longer lifetime and higher connectivity with neighboring nodes, is the major criterion for selecting the best CH. The MO-SHO optimization pseudo code for selecting CHs in L_1 is depicted. Similarly, the CHs are selected for the sensors in L_2. The selected CHs are responsible for collecting the sensed data and assigning scheduling slots. After the election of the CH, it adds the members which are present to its coverage, and it executes member-balanced scheduling (Algorithm 2).

In scheduling sleep and wakeup times, slots are assigned for equal halves of the sensors in the cluster. During wakeup mode, the sensor nodes perform sensing as well as data transmission, whereas, in sleep mode, they never perform any process. The nodes are assigned to sleep mode based on their energy and the number of hops with the CH. The nodes in sleep mode are enabled to mitigate the consumption of energy. When C_1 is a cluster that has a k number of cluster members, $\frac{k}{2}$ sensors are in sleep mode, and the remaining are in awake mode. The energy and number of hops for each member are determined, since they are the most significant constraints that are essential for allotting sleep slots for the sensors.

Algorithm 1 Pseudo Code: MO-SHO Algorithm

Input: Number of Hyenas as Sensors
Output: Best Hyena as CH
1. begin
2. initialize $L_1 = \{s_1, s_2, s_3, \ldots, s_l\}$, /* $l = 1, 2, \ldots i$ */
3. for $(i = 1; i \leq i; i++)$
do
4. initialize parameters $\vec{C}_1$, $\vec{C}_2$, N_H, H /*start optimization*/
5. estimate fitness for each i /*using Equations (5)–(7)*/
6. find best i /*using multiple objective*/
7. while $(I < I_{Max})$ do /*I is the total iterations*/
for i
update position /*using Equation (4)*/
end for
8. update parameters $\vec{C}_1$, $\vec{C}_2$, N_H, H
9. if (i goes beyond search space)
Compute fitness for each i
Update best solution and $\vec{G}_H$

$$I = I + 1$$

End while
10. return best hyena /*selected best CH*/
11. end /*terminate optimization*/

Algorithm 2 Pseudo Code: Member-Balanced Scheduling

Input: total cluster members k
Output: Scheduled time slots
1. begin
2. initialize $k = \{s_1, s_2, s_3, \ldots\}$ /*sensors in a cluster*/
3. for each k
3. find RE and Hp_c /*energy and hop count*/
4. list RE and Hp_c for each k
5.if $(RE < RE_{Th}$&& $Hp_c < Hp_{c(Th)})$ /*comparison with threshold*/
{
assign sleep mode
else
assign wakeup mode
}
end if
6. end

The pseudo code for the member-balanced scheduling is illustrated in sequential steps. As per this scheduling, not all the sensors are put in sleep mode, since all the nodes in sleep mode fail to gather event data if any occur. To avoid this limitation, member-balanced scheduling is performed, in which the node's energy and hop count are determined and then they are split into two equal categories to act in sleep mode and wakeup mode. The sensor in wakeup mode senses the environment and delivers the data to the CH. The data gathered by the CH are sent to the AUV present on their level.

4.3. Data Redundancy Elimination

Data redundancy might occur when inconsistent data duplicates are retransmitted from the sensor nodes; therefore, the existence of duplicate or unnecessary data should be resolved because it can be costly in terms of energy and time consumption. During data gathering by the CH in each cluster, the amount of data can sometimes be large due to the redundant data combined from different sensing nodes in the neighborhood. Thus, the data gathered need to be processed before being transmitted in order to detect and remove redundancy, which can impact the communication traffic and energy consumption of the network in a negative way. We propose an algorithm to measure similarity between the data collected (relative to specific event monitoring) so that an aggregator sensor sends a minimum amount of information and eliminates the duplicate data in a way that means the the latter can deduce the source information of sensing neighbor nodes.

Before transmitting the data, redundancy is minimized by evaluating the similarity between the sensor measurements. The Hassanat distance measure is mathematically formulated as [48]:

$$HasD(s_i, s_j) = \sum_{1}^{k} D(s_i, s_j) \tag{10}$$

$$D(s_i, s_j) = \begin{cases} 1 - \dfrac{1 + min(s_i, s_j)}{1 + max(s_i, s_j)}, & min(s_i, s_j) \geq 0 \\ 1 - \dfrac{1 + min(s_i, s_j) + |min(s_i, s_j)|}{1 + max(s_i, s_j) + |min(s_i, s_j)|}, & min(s_i, s_j) < 0 \end{cases} \tag{11}$$

These distance values are in the range $[0,\ 1]$; a larger value denotes that the data from the two sensors are similar, and, hence, one set of data is eliminated. In this way, the data redundancy is reduced, which also minimizes the amount of gathered data, which, in turn, reduces the energy consumption in data transmission. The CH delivers the collected data only after eliminating the redundancy.

4.4. Data Gathering

Data gathering is the key process in a UWSN performed by an AUV. The data from cluster members are collected by the CH and then delivered to the AUV. The CH is appointed with a waiting time for the arrival of AUV. Once the sensors' data are transmitted and stored at the level of each CH in the different clusters, and in order to minimize the CH energy consumption due to data overload and the data collection delay that can be caused due to a long waiting time for the AUV arrival, this operation must be performed according to pre-defined CH waiting time deadlines. If this waiting time is exceeded, then the CH requests for a route to the gateway and performs inter-cluster routing. A gateway is used as a communication and organization entity with nodes that aim to forward the data between the CHs in different clusters through an optimal, selected route designed in the inter-cluster routing process. Searching for a route with limited knowledge of other CHs tends to consume more energy and cause a delay, so the gateway is employed. A three-step, inter-cluster routing method is incorporated based on the three steps: route discovery, longer route elimination and validation of filtered routes. The gateway, having the knowledge of all the CHs in L_1 and L_2, constructs an undirected graph from which the route is selected towards the AUV. The undirected graph $G = (S, e)$ represents the set of CH sensor nodes S and their edges e. All the possible routes are extracted from G and then the routes with a longer distance and higher hop counts are eliminated. This filtration is performed to ensure faster transmission of data. The increase in distance and hop counts tends to increase end-to-end delay, and, hence, the routes are eliminated in the second step.

Then, a route is selected from the filtered route after validation by the fuzzy–LeNet method. The route validation is handled based on the computation of average energy consumption, fairness, synthesis speed and node efficiency. LeNet is a convolution neural network (CNN) that is combined with fuzzy for accurate results in route validation. The LeNet is composed of a convolutional block and fully connected block. The convolutions block is fed with the input parameters, i.e., a node's average energy consumption, fairness, synthesis speed and node efficiency, based on which, fuzzy membership functions are created.

The LeNet is developed with seven layers: three convolutional layers, two pooling layers, one fully connected layer and one output layer. In the pooling layer, the input parameters computed are crisp values which are converted into the fuzzy set for processing into the fuzzy membership function. In the fully connected layer, each neuron is connected to all the neurons that are present in the next layer. The input for the fully connected layer is the individual parameters of each sensor, which are converted into a single-vector value using the membership function. Then, the output from the fully connected layer is the probability values of each route. The input for the fuzzy–LeNet is the filtered route in which the CHs in the route are computed with the above, discussed parameters. Figure 2 depicts the model of the proposed fuzzy–LeNet for inter-cluster route validation.

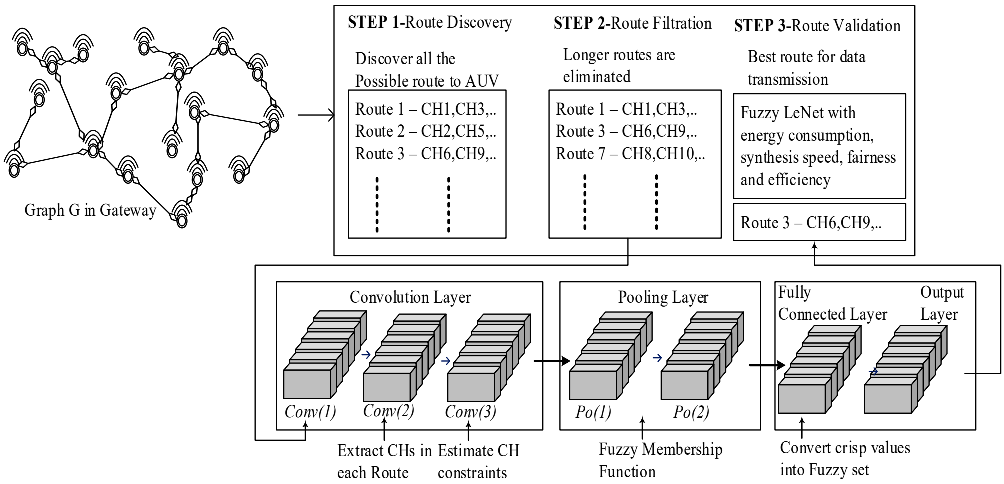

Figure 2. Three-step, inter-Cluster routing with route validation in fuzzy–LeNet.

The average energy consumption of the sensor node is formulated based on its packet transmission with other sensor nodes. The average energy consumption (A_{EC}) of a node is given as:

$$A_{EC} = \frac{EC_{To}}{R_p} \tag{12}$$

Of the terms used above, EC_{To} and R_p are the total energy consumption and received number of packets. The route fairness (RF_s) is mathematically formulated based on Jain's fairness index, which is represented as:

$$RF_s = \frac{\left(\sum_{i=1}^{nCH} tp_i\right)^2}{nCH\left(\sum_{i=1}^{nCH} tp_i^2\right)} \tag{13}$$

This fairness index is formulated for a route in which the number of CHs in a route is nCH, and their throughput is tp_i. The throughput for each CH in the route is involved in fairness prediction. Then, the synthesis speed of the AUV near the CH is predicted using:

$$S_{sp} = \left|cos\left(\beta + arccos\left(\frac{V_f}{V_{AUV}}.cos\beta\right)\right).V_{AUV}/cos\beta\right| \tag{14}$$

$$\beta = arccos\left(\frac{Vc.V_f}{|Vc|.\left|V_f\right|}\right) \tag{15}$$

Here, β is the angle that exists between the CH and the heave velocity V_f, the initial velocity of AUV is represented as V_{AUV} and Vc is the vector between the AUV and the CH. Simply, the synthesis speed is defined as the summation of the heave velocity and the initial velocity of the AUV. Then, node efficiency is determined from the following equation:

$$N_e = 100 * \frac{T_p - RT_p}{T_p} \tag{16}$$

T_p and RT_p are the transmitted number of packets and retransmitted number of packets, respectively. When computing these four parameters for each CH in the route, the convolutional layer is handled and then the values are converted into a fuzzy set in the pooling layer. As a result, the validated best route is preferred for inter-cluster data transmission.

The fuzzy membership functions are modeled as shown in Table 1, with which the route is validated. A set of 16 rules is defined by taking into account the four constraints. The route with higher energy, fairness and efficiency and lower speed of AUV is predicted to be the best route in the validation.

Table 1. Fuzzy membership functions.

A_{EC}	RF_s	S_{sp}	N_e	**Output**
0.5–1	0.5–1	0.5–1	0.5–1	Medium
0.5–1	0.5–1	0.5–1	0–0.5	Low
0.5–1	0.5–1	0–0.5	0.5–1	High
0.5–1	0.5–1	0–0.5	0–0.5	Medium
0.5–1	0–0.5	0.5–1	0.5–1	Low
0.5–1	0–0.5	0.5–1	0–0.5	Very low
0.5–1	0–0.5	0–0.5	0.5–1	Low
0.5–1	0–0.5	0–0.5	0–0.5	Very low
0–0.5	0.5–1	0.5–1	0.5–1	Very low
0–0.5	0.5–1	0.5–1	0–0.5	Very low
0–0.5	0.5–1	0–0.5	0.5–1	Medium
0–0.5	0.5–1	0–0.5	0–0.5	Very low
0–0.5	0–0.5	0.5–1	0.5–1	Low
0–0.5	0–0.5	0.5–1	0–0.5	Low
0–0.5	0–0.5	0–0.5	0.5–1	Low
0–0.5	0–0.5	0–0.5	0–0.5	Very low
0–0.5 indicates low, and 0.5–1 indicates high				

A potential CH forwarder is appointed as suitable if its energy consumption is lower with a high node efficiency in data transmission and higher route fairness and a lower speed of AUV in the area of a CH, because the CHs included in the route are those that exceed the waiting time for the AUV arrival.

This inter-cluster routing is performed in some instances, since the movement of AUV is optimally performed with DACP prediction. This DACP is an intelligent reinforcement learning algorithm that identifies the path by estimating the current ability of the CHs with two factors. A CH, along with its three other one-hop neighboring CHs, is taken for evaluation, and so the data can be collected from these four CHs at one time. This DACP is suitable for performing continuous control tasks, and it concurrently learns the policies. The first and second factor, F_1 and F_2, are estimated as:

$$F_1 = (Bf, (C_T + V_T + W_T)) \tag{17}$$

$$F_2 = (S_s, D_s) \tag{18}$$

The buffer of the CHs is represented as Bf, computation time as C_T, visiting time as V_T, waiting time as W_T, RSSI as S_s and data size as D_s. Using the di-factors, a state value function and soft Q-function are represented as $V_\psi(S_t)$ and $Q_\theta(A_t)$. The default parameters involved in the network are ψ, θ and ϕ; here, the policy in the actor–critic network is given as $\pi_\phi(A_t \middle| S_t)$. The states S_t are defined from the estimated factors F_1 and F_2 of the CH. The state-value function is formulated as:

$$V_\psi(S_t) = \mathbb{E}_{A_t \sim \pi}[Q(S_t, A_t) - log\pi(S_t \middle| A_t)] \tag{19}$$

The π is the policy; here, including an approximate into the state value is not required due to the formulation of the soft-value function. This soft-value function is trained and then the gradient is estimated. Further, the objective is formulated as:

$$J_\pi(\phi) = \mathbb{E}_{S_t \sim D, \epsilon_t \sim \aleph}\left[log\pi_\phi\left(f_\phi(\epsilon_t; S_t)\middle|S_t\right) - Q_\theta\left(S_t, f_\phi(\epsilon_t; S_t)\right)\right] \tag{20}$$

ϵ_t is the given input noise vector, π_ϕ is defined from f_ϕ and D is the sample states and actions that are present in the prior iterations. From this, the approximate gradient $\hat{\nabla}_\phi J_\pi(\phi)$ is expressed as:

$$\hat{\nabla}_\phi J_\pi(\phi) = \nabla_\phi log\pi_\phi(A_t|S_t) + \left(\nabla_{A_t} log\pi_\phi(A_t|S_t) - \nabla_{A_t} Q(S_t, A_t)\right)\nabla_\phi f_\phi(\epsilon_t; S_t \tag{21}$$

The DACP predicts whether the AUV needs to move towards the next position of the CHs. After detecting the location from which the data have to be collected from the four CHs, the mid-point between them is determined and so the AUV can collect data from all the four CHs. The mid-points m_1 and m_2 of the two adjacent CHs are determined, and their mean value is the final position to which the AUV moves. When the CH_a and CH_b are adjacent to each other, the CH_i and CH_j are adjacent to each other so their mid-points m_1 and m_2 are expressed as:

$$m_1 = \left(\frac{x_{1(a)} + x_{2(b)}}{2}, \frac{y_{1(a)} + y_{2(b)}}{2}, \frac{z_{1(a)} + z_{2(b)}}{2}\right) \tag{22}$$

$$m_2 = \left(\frac{x_{1(i)} + x_{2(j)}}{2}, \frac{y_{1(i)} + y_{2(j)}}{2}, \frac{z_{1(i)} + z_{2(j)}}{2}\right) \tag{23}$$

In this 3D-UWSN, the coordinates of CH_a and CH_b are $(x_{1(a)},\ y_{1(a)}, z_{1(a)})$ and $\left(x_{2(b)},\ y_{2(b)}, z_{2(b)}\right)$; similarly, for CH_I and CH_j, the coordinates are $(x_{1(i)},\ y_{1(i)}, z_{1(i)})$ and $\left(x_{2(j)},\ y_{2(j)}, z_{2(j)}\right)$. Further, the moving position is predicted by:

$$Mean = \frac{m_1 + m_2}{2} \tag{24}$$

The prediction of the mean value for the movement of the AUV leads to the collection of data from the four CHs at a time that limits the data collection delay, and, hence, the AUV reaches the CHs within the waiting time.

The data collected from the members are delivered to the AUV during the time of arrival. Based on the predicted mid-point, the AUV is exactly positioned in between the four CHs at a one-hop distance and collects the data as shown in Figure 3. If the AUV's arrival to the next position is delayed, then the collected data are transmitted by the CH via inter-cluster routing.

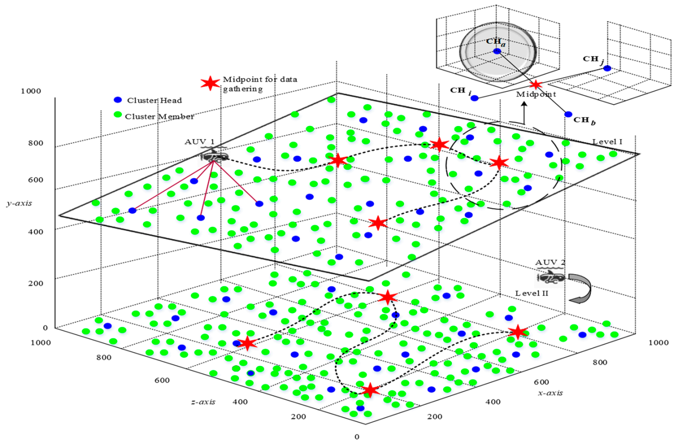

Figure 3. Movement of AUV in 3D-UWSN.

4.5. Emergency Event Transmission

The 3D-UWSN is an environment in which the sensors give current, collected data that can either be critical or non-critical. In the case of a critical, sensed message, the occurrence of the event is predicted, and, hence, it is forwarded by selecting a forwarder using the weighted method. The weighted value for neighboring CHs is computed and then a higher-weighted CH is selected to forward the emergency event message. The constraints that are taken into account are distance, load and residual energy. The Euclidean distance is presented to determine the distance between the two CHs, and it is expressed for CH_i and CH_j:

$$D_{(CH_i,CH_j)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{25}$$

The sensor nodes are deployed in 3D space, and so the coordinates of the CHs CH_i and CH_j are indicated as (x_1, y_1, z_1) and (x_2, y_2, z_2). Along with this distance, the remaining energy RE and the load at the CH L_{CH_i} is determined as a weight value W_{CH_i}.

$$W_{CH_i} = \left(w_1 * D_{(CH_i,CH_j)}\right) + (w_2 * RE) + \left(w_3 * L_{CH_i}\right) \tag{26}$$

From the estimated weight, the forwarder is selected, and the data are transmitted. The load depends on the available packets that are yet to be transmitted from the CH. Here, the weights $w_1 + w_2 + w_3 = 1$ are constant. Once the CH with high weight is selected as forwarder, then the emergency event is forwarded and reaches the AUV.

5. Performance Evaluation

In this section, the testing environment of the proposed DEDG 3D-UWSN is studied along with the results achieved by previous research works. This section is divided into three parts: simulation environment, comparative analysis and research highlights. The better efficiency of the proposed system is elaborated with graphical justification in this section.

5.1. Simulation Environment

This 3D-UWSN was implemented in Network Simulation Ns3.26, which is suitable for designing a 3D-UWSN. AquaSim is one of the most important modules in Ns3 for simulating the underwater sensor environment, and it is supplemented by other modules for network model creation. However, this is a simulation environment; it supports with important network modules that enable the creation of a real-world environmental setup. The simulation parameters are not limited in any way. The simulation was carried out using Ubuntu 14.04 LTS with a 32-bit CPU.

The parameters that were used in the DEDG 3D-UWSN simulation are illustrated in Table 2. These are the significant specifications, but these parameters were not limited. The Ns3.26 was installed onto an Ubuntu 14.04 operating system, which was supported on a 32-bit system. Ns3.26 simulator is flexible for designing a 3D-UWSN with the processes of clustering, scheduling, inter-cluster routing, event transmission and the movement of the AUV. As discussed in previous section, each process was handled as per the designed algorithms.

According to the simulation environment, the process in this research was performed as shown in Figure 4. As per the simulation, once the data arecollected from all the CHs in a level, then the simulation is completed. Meanwhile, the performance of this simulation setup is evaluated in terms of performance metrics.

Table 2. Simulation setup in Ns3.

Parameter		Specification
3D-UWSN entities	Simulation area	1000 × 1000 × 1000
	Number of underwater sensors	100
	Number of sensors in level I	50
	Number of sensors in level II	50
	Number of AUVs	2
	Number of gateways	1
	Number of surface sinks	1
	Number of clusters in each level	5–7
	Simulation time	300 s
	Modules used	AquaSim, Antenna, Config Store, CSMA, LTE, AODV, Mesh, Mobility, DSR, Flow Monitor and Internet
Underwater sensor parameters	Packet size	512 kb
	Total number of packets	200
	Packet time interval	100 ms
	Data rate	10–20 Mbps
	Initial energy per sensor	100 J
	Transmission range	400 m

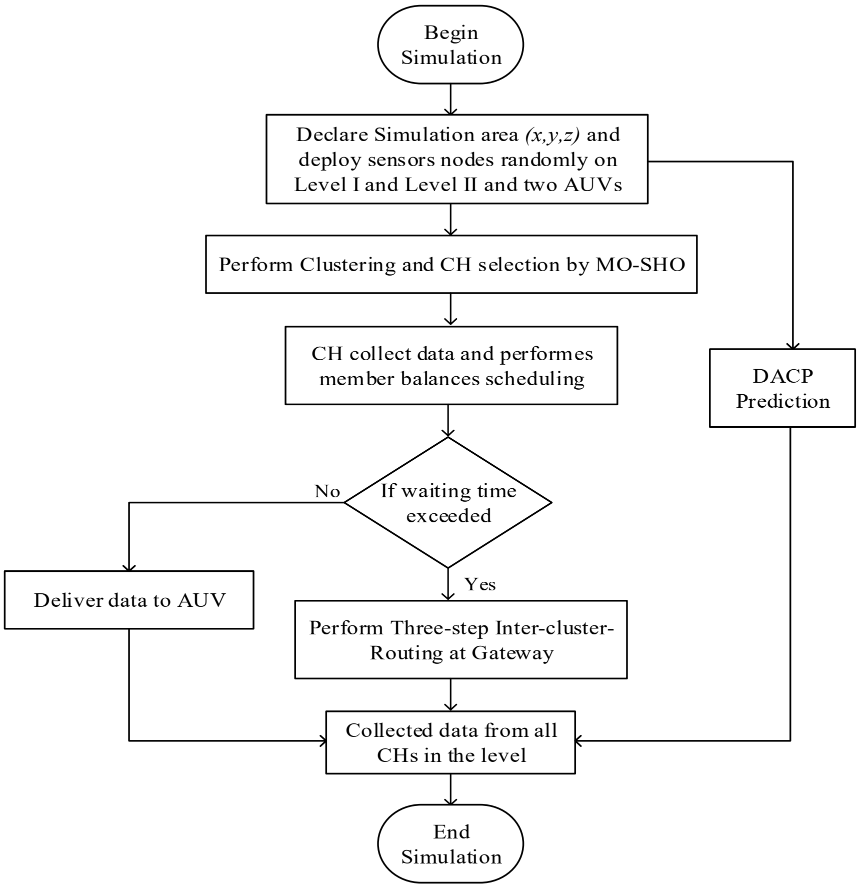

Figure 4. Simulation workflow of DEDG 3D-UWSN.

The Ns3 simulation environment was programmed in C++ language, and Python was used for compiling the designed DEDG 3D-UWSN. The result obtained in Ns3 was executed, and the visualization is shown in Figure 5. In this network, the proposed algorithms were applied, and the results were evaluated.

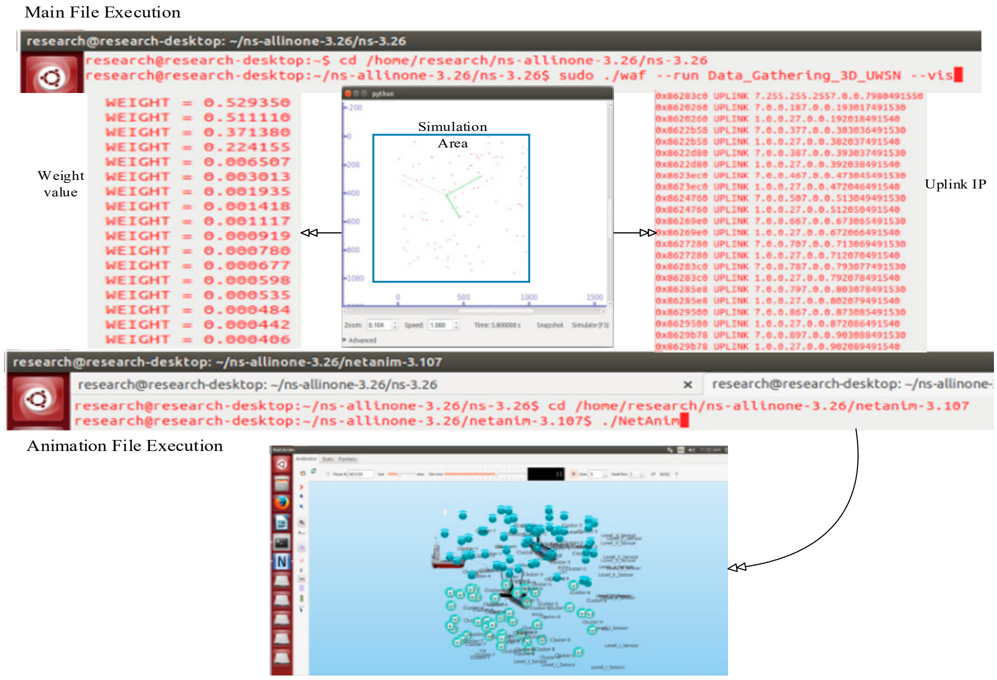

Figure 5. Ns3 Setup for DEDG 3D-UWSN.

As shown in Figure 6, the proposed DEDG 3D-UWSN design is suitable for a variety of real-world applications; here, specifically, it is demonstrated for an underwater pipeline monitoring system. The underwater pipelines are monitored by the deployed underwater sensors. The mobile AUV collects data from the sensors and delivers it to the surface sink. In case of any leakage in the pipeline, it is measured by the sensors and immediately reported to the AUV for faster recovery. Since this work supports normal data collection with limited delay and energy consumption, it is applicable for other applications.

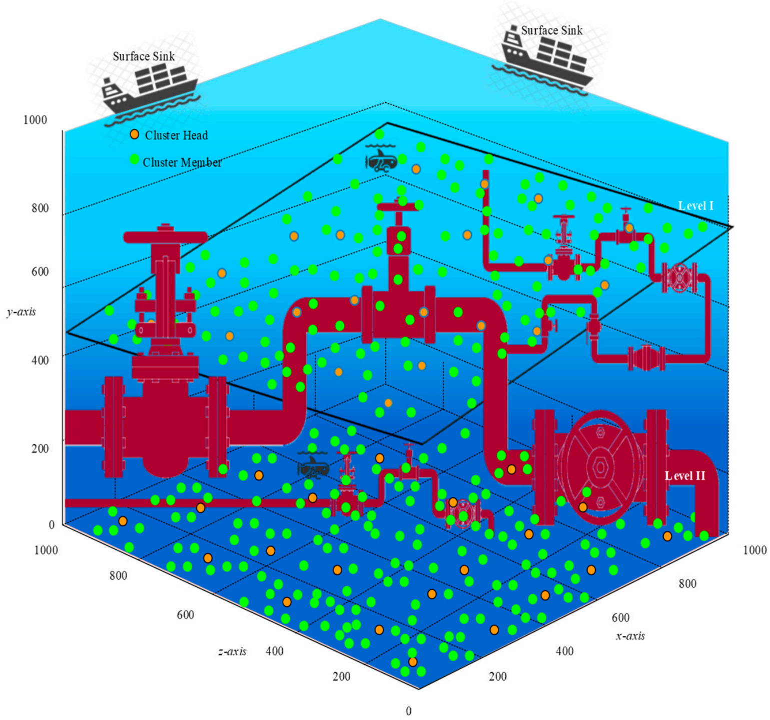

Figure 6. Application scenario.

5.2. Comparative Analysis

The comparative analysis was incorporated with the key objective of minimization of energy and delay in the network. Energy consumption and delay are the two challenging constraints in sensor nodes that are required to be as low as possible during the transmission of data. In a UWSN, the common processes performed by the sensor are sensing and transmitting the sensed data to the AUV and then to the sink. Here, the data collecting AUV is not subjected to limited energy, and, hence, it moves in and around the network to collect the data from sensors.

The efficient solution of using an AUV for data collection is concentrated in the UWSN environment to ensure limited energy consumption when gathering data. Table 3 discusses the prior research work in which the underwater sensor data were collected from nodes in the network. The existence of demerits in each work is illustrated, and it is resolved in this proposed work, which depicts on output result. The comparative parameters that are taken in account are based on the significance of the network lifetime and the delay that exists during data transmission and data gathering. The better efficiency of this proposed system was estimated and compared with the previous research works on UWSNs that concentrated on data gathering with AUVs.

Table 3. Comparison on AUV-based data gathering.

Work	Process and # of AUVs	Demerits
[43]	AEC mechanism 1. AUV	• CH is selected in each virtual sector by using energy; cases which consider only this metric have the possibility to select a node as head with a smaller degree and located at farthest distance. This degrades the data gathering efficiency; • Inappropriate criteria (delay and transmission time) for scheduling.
[44]	ALP 1. AUV	• Use of pre-defined spiral path increases AUV's tour length, which eventually increases data collection delay; • Single AUV increases waiting time at sensors, which leads to drop in sensed information due to its capacity limitation.
[45]	Bipartite K-means 1. AUV	• The cluster construction time is large since, initially, the clustering is performed with random centers and then, again, the cluster centers are reselected based on the distance metric; • The path prediction of an AUV is based on the determination of the shortest data collection time; however, use of a single AUV increases traveling distance, which eventually extends delay. In case of an event message, it fails to deliver the message promptly.
[46]	ELMPP 2. AUVs	• More energy consumption in the relay node due to frequent selection of the same sensor as relay. This happens because the fixed path in the AUV is based on line segments; • RSSI is measured to select relay, which is not efficient since RSSI gives only the strength of the communication link but not the strength of the node.

The previous works on UWSNs dealt with AUVs that move underwater to collect sensed data from the sensors. In previous works, the sensed data were directly collected from individual sensor nodes or from the CHs. The supportability for event message transmission in the UWSNs was not considered in previous works, as shown in Table 4. The processes of clustering, scheduling and AUV path prediction are performed for data gathering. The main goal of this UWSN was to gather data with limited energy consumption and delay. The possible solutions that were involved to prolong network lifetime and minimize delay are:

- To develop clusters in the network reduces energy consumption, since the data is collected by a CH, and it delivers to the AUV;
- The assignment of sleep slots for sensor nodes enables the reduction of energy consumption in sensors due to the nil processing for certain time period;
- Appropriate path planning of AUV minimizes waiting time in CHs, and that reduces the delay in gathering.

Table 4. Comparison with existing works.

Work	Clustering	Scheduling	AUV Path Prediction	Emergency Event Transmission
[43]	✓	✓	✓	×
[44]	×	×	✓	×
[45]	✓	×	✓	×
[46]	×	×	✓	×
DEDG 3D-UWSN	✓	✓	✓	✓

✓—discussed, ×—not discussed

The efficiency of the proposed DEDG 3D-UWSN was evaluated by measuring the network lifetime in terms of energy consumption and number of alive sensors. Then, the data gathering efficiency was depicted by the estimation of packet delivery ratio, end-to-end delay, collection delay and tour length.

5.2.1. Efficiency of Network Lifetime

The energy efficiency of the proposed DEDG 3D-UWSN was compared with previous works on AEC, ALP, bipartite K-means and ELMPP processes. In ALP and ELMPP, the clustering is not performed, and, hence, each node delivers its data individually to the AUV. Even though scheduling is present in AEC, it is not defined based on the energy constraint of the sensor node. The existence of these limitations degrades the network lifetime. The energy consumption increases due to longer waiting time, multiple hops in data transmission, continuous sensing and elimination of energy-aware constraints. Figure 7 illustrates the comparison with previous works relating to energy consumption, and the results show the DEDG 3D-UWSN has lower energy consumption than the other, previous works.

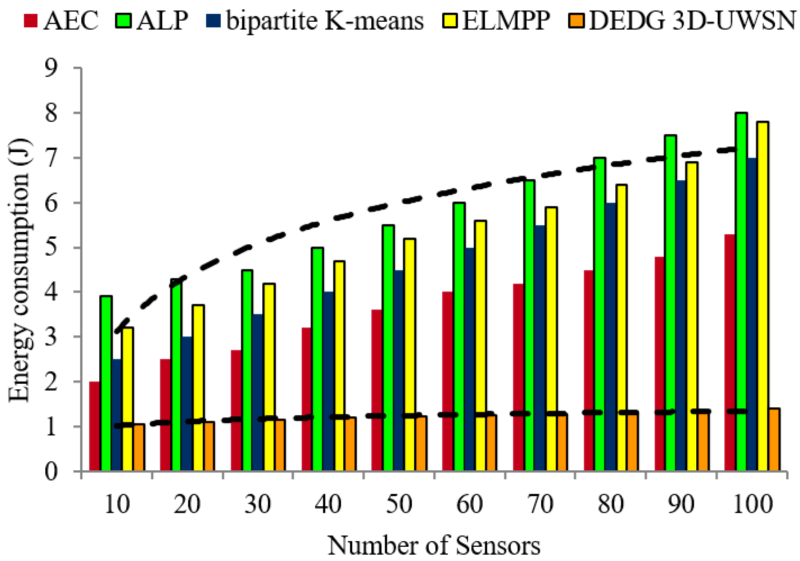

Figure 7. Comparison of energy consumption.

The reduction in energy consumption tends to increase the number of alive sensor nodes in the network. In the DEDG 3D-UWSN, the energy consumption is minimized; thereby, it increases the number of alive sensors and extends network lifetime, providing the following solutions:

- Clustering with optimal CH selection enhances the stability of the CH, and this minimizes frequent selection of the CH as well as cluster formation. Without the knowledge of neighboring nodes, the cluster cannot be constructed, which requires exchange of hello packets with the neighboring nodes, which consumes energy;

- Scheduling the sensor with sleep slots based on their current energy status is the major reason for keeping the sensor alive for multiple rounds;
- The route selected by the CH itself consumes energy, and, hence, a gateway is used for predicting a route that reduces energy consumption.

The number of alive nodes is illustrated in Figure 8, which shows that ALP has the worst performance, and DEDG 3D-UWSN has the best performance among all the works. As per this comparison, the increase in simulation time also increases the amount of transmission by the sensor nodes. In total, 100 sensors participated in the network, of which 91 nodes were active after 300 s, whereas only 75 nodes were active in AEC. Other works relating to ALP, bipartite K-means and ELMPP were able to maintain 60–75 sensors at the end of simulation.

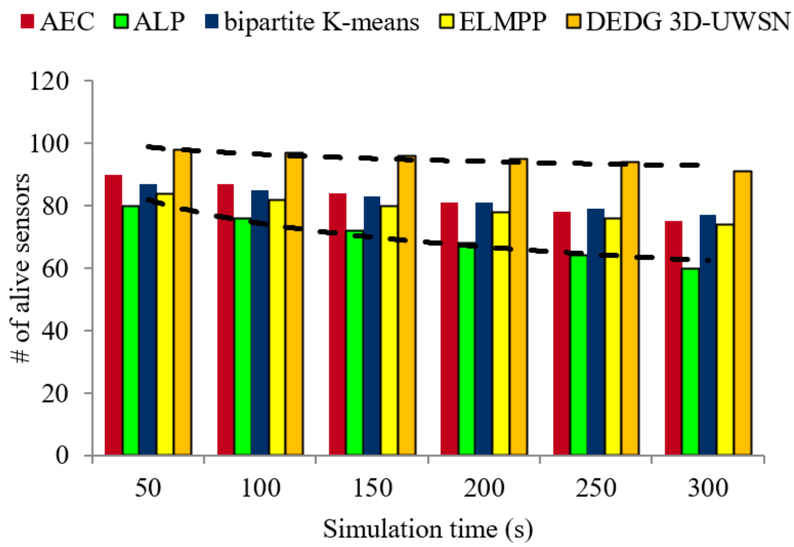

Figure 8. Comparison of alive nodes.

The number of dead nodes is lower in the DEDG 3D-UWSN than in the other works, as shown in Table 5. From this table, it can be seen that bipartite K-means has a lower number of dead nodes than the proposed work, and this is due to the performance of the clustering algorithm. However, the cluster construction in bipartite K-means consumes time, it constructs a cluster and then the data are gathered.

Table 5. Comparison of dead sensors.

Simulation Time	# of Dead Sensor Nodes				
	AEC	ALP	Bipartite K-Means	ELMPP	DEDG 3D-UWSN
100	13	24	15	18	3
200	19	32	19	22	5
300	25	40	23	26	9

Network lifetime was evaluated with respect to the number of sensors, and the results are shown in Figure 9. The higher network lifetime reflects the efficiency of the proposed DEDG 3D-UWSN. As an average, the previous works relating to ALP, ELMPP and bipartite graph achieved 80–88% network lifetime, whereas 97% was reached in the DEDG 3D-UWSN. Therefore, the processes performed in this work are efficient, and, hence, there is improvement in the network lifetime.

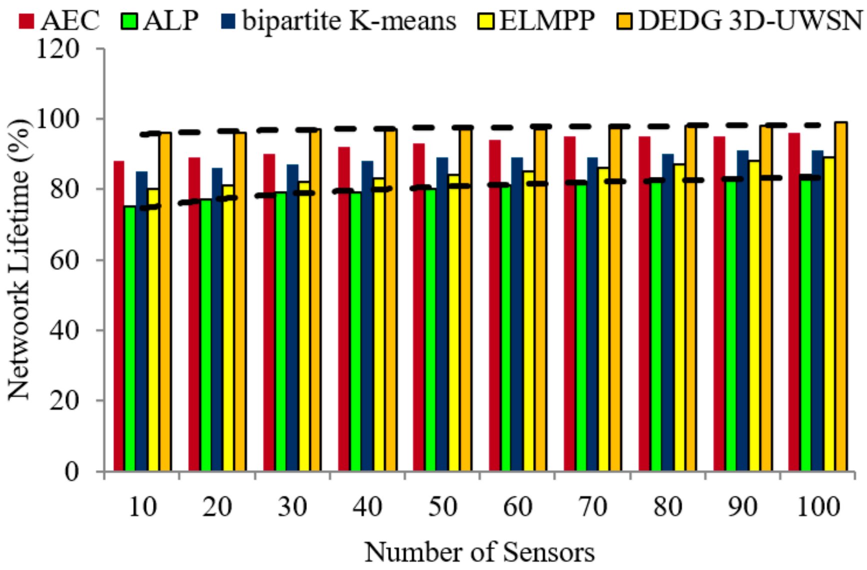

Figure 9. Comparison of network lifetime.

5.2.2. Efficiency of Data Gathering

The efficiency of data gathering is the key idea that is addressed by the AUV. The AUV is enabled to predict the path and gather data from the employed sensors. The effectiveness of data gathering is estimated in terms of packet delivery ratio, end-to-end delay and collection delay. Packet delivery ratio is a significant metric that determines the success of data transmission via the selected route.

Figure 10 indicates that the DEDG 3D-UWSN increases packet delivery ratio when compared with previous research works. This is due to the prediction of a route after validation by a gateway, and, hence, no packet is dropped or lost. The selection of neighbors in the route is required to be appropriate for the data transmission, else the packet delivery ratio is degraded. The previous ALP, bipartite K-means and ELMPP processes achieved 83%, 91% and 86%, respectively. Comparatively, AEC reached 95%, and DEDG 3D-UWSN reached 98% of packet delivery ratio. On the other hand, delay was also measured, since the data were gathered by the appointed AUV. The end-to-end delay defines the time taken to transmit the collected data from a CH to the AUV via intermediate hops. The collection delay depends on the time taken by the AUV to collect data from the set of sensors.

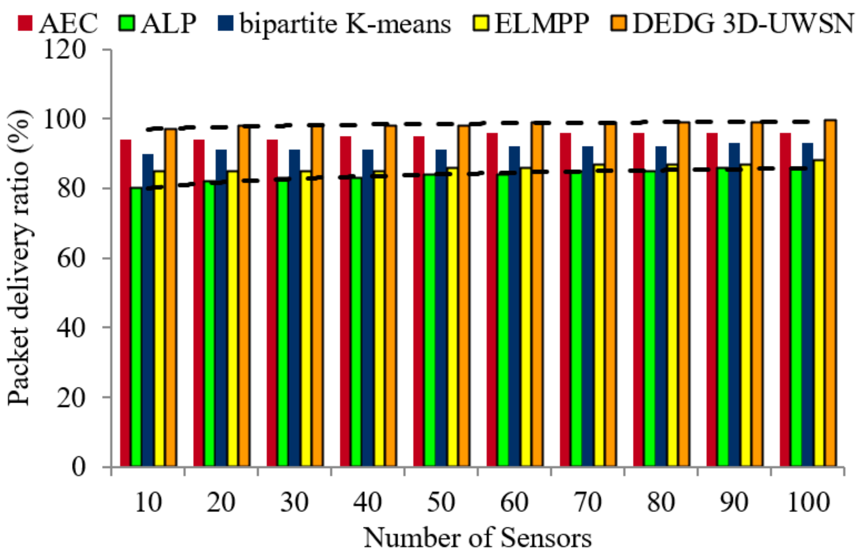

Figure 10. Comparison of packet delivery ratios.

The performances of end-to-end delay and collection delay are illustrated in Figures 11 and 12, respectively. End-to-end delay is measured with respect to the number of hop counts and the collection delay with respect to the increase in the number of sensors in the network. The result shows that both delays were lower than in the previous works. This reduction is due to the efficient path prediction in the AUV and the inter-cluster routing, which validates the route using fuzzy–LeNet. In previous works, the AUV's path was planned and based on the data collected from sensors. The AUV collected sensed data only from a single sensor at a time, whereas, in the DEDG 3D-UWSN, the data are collected from four CHs at a time. Due to this, the AUV is enabled to collect data in the path faster, which reduces collection delay. In contrast, ELMPP has the highest collection delay, since it collects data from relay one after the other. The absence of clustering is also reflected in the increases in collection delay, since the data sensed by the sensor have to wait a longer time for the arrival of the AUV to collect their data.

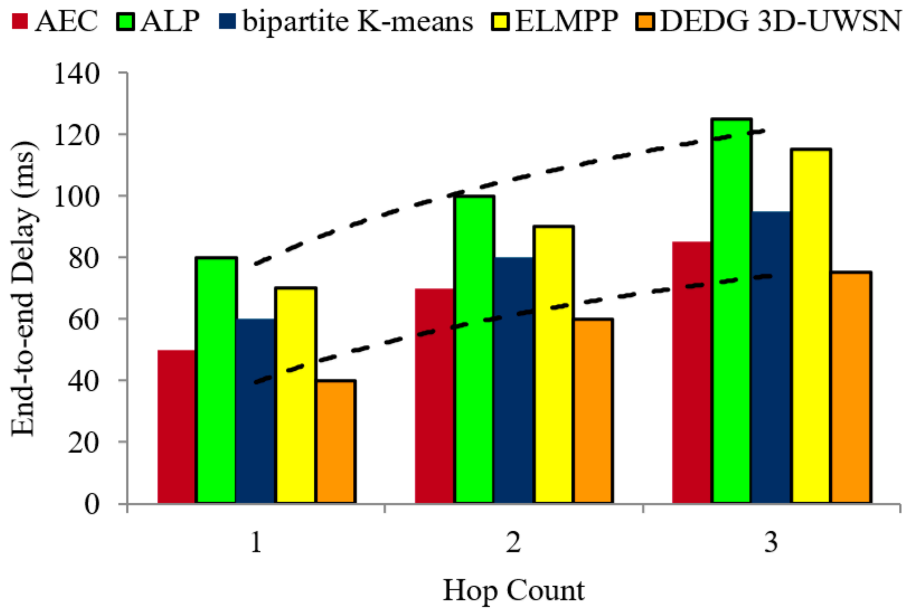

Figure 11. Comparison of end-to-end delay.

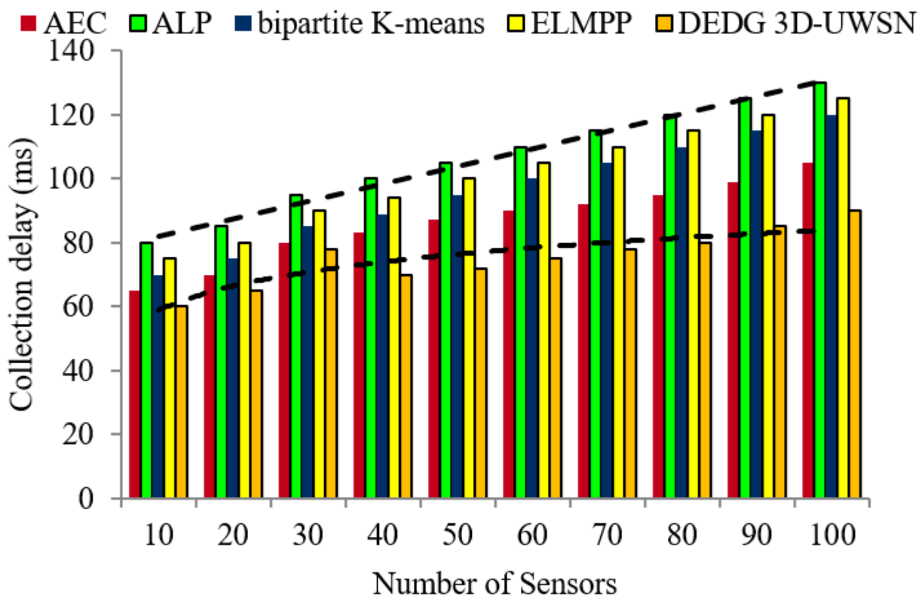

Figure 12. Comparison of collection delay.

The AUV tour length also impacts and increases the collection delay. If the tour length is longer, then the data collected by the sensor node have to hold on the packets until the AUV approaches near the sensor. Only the appropriate planning of the path in the AUV can minimize the tour length.

In the previous works, the authors also discussed the assignment of a pre-defined path for data collection. Table 6 illustrates the comparison of the tour length of the AUV in each work. The tour length was reduced by 30–50 min in the proposed method compared to the AEC method. From this comparative analysis, the better efficiency of the proposed DEDG 3D-UWSN is illustrated, and the improvement in performance is due to the process handled in this work. The processes of clustering and scheduling aim to save energy in sensor nodes, and data gathering is performed with an AUV, the path of which is predicted effectively.

Table 6. AUV tour length.

# of Sensors	Tour Length (m)				
	AEC	ALP	Bipartite K-Means	ELMPP	DEDG 3D-UWSN
20	350	450	390	420	320
40	400	490	420	450	350
60	420	530	470	490	370
80	470	570	520	540	420
100	500	630	580	600	450

5.3. DEDG 3D-UWSN Highlights

The designed DEDG 3D-UWSN was developed to effectively collect data from the network nodes with the aim of improving network lifetime and reducing delay in transmission. The idea of this work was to present support for both normal data gathering, as well as emergency event transmission. The major process handled in this DEDG 3D-UWSN is depicted in Table 7. This shows that the process, along with the resulting improvement that is achieved by our simulation, is theoretically justified.

Table 7. Process involved in DEDG 3D-UWSN.

Process	Resulting Improvement
Clustering	Reduces energy consumption in data gathering from the CHs
Optimal CH selection	Reduces energy consumption by mitigating unnecessary selection of CH which needs to exchange node information
Sleep–wakeup scheduling	Saves energy of the sensor, and also all the data are sensed
Data gathering	Reduces energy consumption since the AUV is positioned one hop from the CH Reduced collection delay by collecting data from four CHs at a time

As discussed in earlier sections, the proposed DEDG 3D-UWSN performs data gathering with the assistance of two AUVs in the network and achieves prolonged network lifetime and minimizes delay in data collection. Here, we provide the major research highlights of this DEDG 3D-UWSN:

- The 3D-UWSN with two AUVs incorporated was designed with the aim of achieving delay-aware data gathering by optimal positioning of AUVs and by using inter-cluster routing in case of exceeded waiting time. To assist a large-scale network and faster data gathering, two AUVs are employed. Incorporation of sleep time for sensor nodes reduces energy consumption;
- The optimal selection of CH using MO-SHO ensures prolonged sustainment of CH, and the reduction of redundant data improves delivery time. Then, data gathering of the AUV from for CHs at optimal position reduces delay. Here, fuzzy–LeNet was used, which performs faster and results in an appropriate solution;

- In conditions with delayed arrival of the AUV, the gathered data are transmitted to the AUV via an inter-cluster route. This is performed in order to make free space in CHs to gather upcoming sensed information from CHs.

The sensors not only sense normal data; they also sense critical emergency data which are sent immediately by selecting a forwarder which sequentially reaches the AUV and then the surface sink

6. Conclusions

This paper developed a proposed DEDG 3D-UWSN with the aim of improving the network lifetime and reducing the delay. The key processes handled in this work to address this aim were: clustering with MO-SHO for optimal CH selection; member-balanced scheduling to allot sleep slots; three-step, inter-cluster routing with fuzzy–LeNet; DACP prediction in AUVs; and event message transmission. Initially, the 3D-UWSN environment is categorized into two levels relating to the depth of the ocean. The sensors in each level are clustered, and an optimal CH is selected by estimating the fitness with node's lifetime, degree and centrality. The optimal selection of CH results in the sustainability of the CH for a longer time, and it reduces the energy consumption for frequent CH selection. An equal number of members in the cluster is assigned to sleep mode and wakeup mode by energy consumption and the number of hops from the CH. The CH also reduces redundant data by measuring the Hassanat distance and eliminates similar data before forwarding it to the AUV. According to this proposed work, the CHs wait for a particular waiting time unless the AUV reaches a CH, then they request the gateway for an inter-cluster path. For path selection, the gateway identifies all the possible routes, then ignores longer routes, and, further, the filtered routes are validated using fuzzy–LeNet by considering energy consumption, fairness, synthesis speed and efficiency. This inter-cluster routing occurs in rare cases, since the movement of the AUV is predicted by two factors. These CH parameters are fed into DACP prediction followed by the estimation of the mid-point and then the AUV moves to collect data from the four CHs. Therefore, this reduces delay in data collection. This work also supports the transmission of event messages using a weighted method which selects a forwarder towards the AUV. The proposed DEDG 3D-UWSN showed better results in terms of energy efficiency and delay when compared with previous works.

In future, this DEDG 3D-UWSN architecture is planned to be extended with network partitioning for further reduction of the energy consumption of sensors nodes. additionally, this environment will be tested in any specific application that exists for UWSNs.

Author Contributions: Conceptualization, R.A. and A.C.; methodology, M.S.A.M.; software, R.A.; validation, A.M. and A.C.; formal analysis, R.A.; investigation, M.A.A.; resources, N.A.S.; data curation, M.A.A.; writing—original draft preparation, A.C. and M.S.A.M.; writing—review and editing, D.P. and A.C.; visualization, A.M.; supervision, A.M.; project administration, R.A.; funding acquisition, D.P. and R.A. All authors have read and agreed to the published version of the manuscript.

Funding: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Part of the reported study was funded by RFBR, project number 20-37-90131 (recipient Dmitry Poluektov).

Data Availability Statement: The data are contained within the article and/or available from the corresponding author upon reasonable request.

Acknowledgments: The authors express their gratitude to Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

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

References

1. Panchal, H.; Gajjar, S. Fuzzy Logic-Based Cluster Head Selection an Underwater Wireless Sensor Network: A Survey. In *Communication and Intelligent Systems. Lecture Notes in Networks and Systems*; Sharma, H., Shrivastava, V., Kumari Bharti, K., Wang, L., Eds.; Springer: Singapore, 2022; Volume 461. [CrossRef]
2. Islam, K.Y.; Ahmad, I.; Habibi, D.; Waqar, A. A survey on energy efficiency in underwater wireless communications. *J. Netw. Comput. Appl.* **2022**, *198*, 103295. [CrossRef]
3. Watt, A.J.; Phillips, M.R.; Campbell, C.E.; Wells, I.; Hole, S. Wireless Sensor Networks for monitoring underwater sediment transport. *Sci. Total Environ.* **2019**, *667*, 160–165. [CrossRef]
4. Yang, G.; Dai, L.; Wei, Z. Challenges, Threats, Security Issues and New Trends of Underwater Wireless Sensor Network. *Sensor* **2018**, *18*, 3907. [CrossRef]
5. Sharma, M.; Gupta, A.; Gupta, S.K.; Alsamhi, S.H.; Shvetsov, A.V. Survey on Unmanned Aerial Vehicle for Mars Exploration: Deployment Use Case. *Drones* **2022**, *6*, 4. [CrossRef]
6. Chaaf, A.; Saleh Ali Muthanna, M.; Muthanna, A.; Alhelaly, S.; Elgendy, I.A.; Iliyasu, A.M.; El-Latif, A.; Ahmed, A. Energy-efficient relay-based void hole prevention and repair in clustered multi-AUV underwater wireless sensor network. *Secur. Commun. Netw.* **2021**, *2021*, 9969605. [CrossRef]
7. Saif, A.; Dimyati, K.; Noordin, K.A.; Shah, N.S.M.; Alsamhi, S.H.; Abdullah, Q. Energy-Efficient Tethered UAV Deployment in B5G for Smart Environments and Disaster Recovery. In Proceedings of the 2021 1st International Conference on Emerging Smart Technologies and Applications (eSmarTA), Sana'a, Yemen, 10–12 August 2021; pp. 1–5. [CrossRef]
8. Bansal, R.; Maheshwari, S.; Awwal, P. Energy-efficient Multilevel Clustering Protocol for Underwater Wireless Sensor Networks. In Proceedings of the 9th International Conference on Cloud Computing, Data Science & Engineering, Uttar Pradesh, India, 10–11 January 2019.
9. Alsamhi, S.H.; Shvetsov, A.V.; Kumar, S.; Hassan, J.; Alhartomi, M.A.; Shvetsova, S.V.; Sahal, R.; Hawbani, A. Computing in the Sky: A Survey on Intelligent Ubiquitous Computing for UAV-Assisted 6G Networks and Industry 4.0/5.0. *Drones* **2022**, *6*, 177. [CrossRef]
10. Cheng, E.; Wu, L.; Yuan, F.; Gao, C.; Yi, J. *Node Selection Algorithm for Underwater Acoustic Sensor Network Based on Particle Swarm Optimization*; IEEE Access: Piscataway, NJ, USA, 2019; Volume 7, pp. 164429–164443.
11. Bhattacharjya, K.; Alam, S.; De, D. CUWSN: Energy efficient routing protocol selection for cluster based underwater wireless sensor network. *J. Microsyst. Technol.* **2019**, *28*, 543–559. [CrossRef]
12. Su, R.; Zhang, D.; Li, C.; Gong, Z.; Venkatesan, R.; Jiang, F. Localization and Data Collection in AUV-Aided Underwater Sensor Networks: Challenges and Opportunities. *IEEE Netw.* **2019**, *33*, 86–93. [CrossRef]
13. Islam, T.; Lee, Y.K. A Comprehensive Survey of Recent Routing Protocols for Underwater Acoustic Sensor Networks. *Sensors* **2019**, *19*, 4256. [CrossRef]
14. Gomathi, R.M.; Manickam, J. Energy Efficient Static Node Selection in Underwater Acoustic Wireless Sensor Network. *Wirel. Pers. Commun.* **2019**, *107*, 709–727. [CrossRef]
15. Alsamhi, S.H.; Shvetsov, A.V.; Kumar, S.; Shvetsova, S.V.; Alhartomi, M.A.; Hawbani, A.; Rajput, N.S.; Srivastava, S.; Saif, A.; Nyangaresi, V.O. UAV Computing-Assisted Search and Rescue Mission Framework for Disaster and Harsh Environment Mitigation. *Drones* **2022**, *6*, 154. [CrossRef]
16. Xiao, L.; Liang, Y.; Weng, C.; Yang, D.; Zhao, Q. Uavenabled data collection: Multiple access, trajectory optimization, and energy trade-off. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 9647539. [CrossRef]
17. Mahmoudzadeh, S.; Powers, D.M.; Atyabi, A. UUV's Hierarchical DE-Based Motion Planning in a Semi Dynamic Underwater Wireless Sensor Network. *IEEE Trans. Cybern.* **2018**, *49*, 2992–3005. [CrossRef] [PubMed]
18. Khisa, S.; Moh, S. Survey on recent advancements in energy-efficient routing protocols for underwater wireless sensor networks. *IEEE Access* **2021**, *9*, 55045–55062. [CrossRef]
19. Alsamhi, S.H.; Shvetsov, A.V.; Shvetsova, S.V.; Hawbani, A.; Guizan, M.; Alhartomi, M.A.; Ma, O. Blockchain-Empowered Security and Energy Efficiency of Drone Swarm Consensus for Environment Exploration. *IEEE Trans. Green Commun. Netw.* **2022**, *11*, 2400–2473. [CrossRef]
20. Li, M.; Du, X.; Liu, X.; Li, C. Shortest Path Routing Protocol Based on the Vertical Angle for Underwater Acoustic Networks. *J. Sens.* **2019**, *2019*, 9145675. [CrossRef]
21. Roberge, V.; Tarbouchi, M.; Labonté, G. Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time UAV Path Planning. *IEEE Trans. Ind. Inform.* **2013**, *9*, 132–141. [CrossRef]
22. MahmoudZadeh, S.; Yazdani, A.M.; Sammut, K.; Powers, D.M. Online path planning for AUV rendezvous in dynamic cluttered undersea environment using evolutionary algorithms. *Appl. Soft Comput.* **2018**, *70*, 929–945. [CrossRef]
23. Felemban, E.; Shaikh, F.K.; Qureshi, U.M.; Sheikh, A.A.; Qaisar, S.B. Underwater Sensor Network Applications: A comprehensive Survey. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 896832. [CrossRef]
24. Luo, J.; Fan, L.; Wu, S.; Yan, X. Research on Localization Algorithms Based on Acoustic Communication for Underwater Sensor Networks. *Sensors* **2018**, *18*, 67. [CrossRef]
25. Wang, S.; Shin, Y. Efficient Routing Protocol Based on Reinforcement Learning for Magnetic Induction Underwater Sensor Networks. *IEEE Access* **2019**, *7*, 82027–82037. [CrossRef]

26. Ghoreyshi, S.M.; Shahrabi, A.; Boutaleb, T.; Khalily, M. Mobile Data Gathering with Hop-Constrained Clustering in Underwater Sensor Networks. *IEEE Access* **2019**, *7*, 21118–21132. [CrossRef]
27. Li, J.; Zhang, J.; Zhang, G.; Zhang, B. An Adaptive Prediction Target Search Algorithm for Multi-AUVs in an Unknown 3D Environment. *Sensors* **2018**, *18*, 3853. [CrossRef] [PubMed]
28. Chang, H.; Feng, J.; Duan, C. Reinforcement Learning-Based Data Forwarding in Underwater Wireless Sensor Networks with Passive Mobility. *Sensors* **2019**, *19*, 256. [CrossRef]
29. Bouabdallah, F. Selection of Optimal Deployment and Routing Configurations in Underwater Acoustic Sensor Networks for Avoiding Energy Holes. *Mob. Inf. Syst.* **2019**, *2019*, 3597304. [CrossRef]
30. Wadud, Z.; Ismail, M.; Qazi, A.B.; Khan, F.A.; Derhab, A.; Ahmad, I.; Ahmad, A.M. An Energy Balanced Efficient and Reliable Routing Protocol for Underwater Wireless Sensor Networks. *IEEE Access* **2019**, *7*, 175980–195999. [CrossRef]
31. Hindu, S.K.; Hyder, W.; Luque-Nieto, M.A.; Poncela, J.; Otero, P. Self-Organizing and Scalable Routing Protocol (SOSRP) for Underwater Acoustic Sensor Networks. *Sensors* **2019**, *19*, 3130. [CrossRef]
32. Khalid, M.; Ahmad, F.; Arshad, M.; Khalid, W.; Ahmad, N.; Cao, Y. E2MR: Energy-efficient multipath routing protocol for underwater wireless sensor networks. *IET Netw.* **2019**, *8*, 321–328. [CrossRef]
33. Khan, W.; Wang, H.; Anwar, M.S.; Ayaz, M.; Ahmad, S.; Ullah, I. A Multi-Layer Cluster Based Energy Efficient Routing Scheme for UWSNs. *IEEE Access* **2019**, *7*, 77398–77410. [CrossRef]
34. Ahmed, G.; Zhao, X.; Fareed MM, S.; Fareed, M.Z. An Energy-Efficient Redundant Transmission Control Clustering Approach for Underwater Acoustic Networks. *Sensors* **2019**, *19*, 4241. [CrossRef]
35. Faheem, M.; Tuna, G.; Gungor, V.C. QERP: Quality-of-Service (QoS) Aware Evolutionary Routing Protocol for Underwater Wireless Sensor Networks. *IEEE Syst. J.* **2018**, *12*, 2066–2073. [CrossRef]
36. Durrani, M.Y.; Tariq, R.; Aadil, F.; Maqsood, M.; Nam, Y.; Muhammad, K. Adaptive Node Clustering Technique for Smart Ocean under Water Sensor Network (SOSNET). *Sensor* **2019**, *19*, 1145. [CrossRef]
37. Jawhar, I.; Mohamed, N.; Al-Jaroodi, J.; Zhang, S. An Architecture for Using Autonomous Underwater Vehicles in Wireless Sensor Networks for Underwater Pipeline Monitoring. *IEEE Trans. Ind. Inform.* **2019**, *15*, 1329–1340. [CrossRef]
38. Yan, Z.; Li, J.; Wu, Y.; Zhang, G. A Real-Time Path Planning Algorithm for AUV in Unknown Underwater Environment Based on Combining PSO and Waypoint Guidance. *Sensors* **2018**, *19*, 20. [CrossRef] [PubMed]
39. Chen, M.; Zhu, D. Data collection from underwater acoustic sensor networks based on optimization algorithms. *Computing* **2020**, *102*, 83–104. [CrossRef]
40. Wang, X.; Wei, D.; Wei, X.; Cui, J.; Pan, M. HAS4: A Heuristic Adaptive Sink Sensor Set Selection for Underwater AUV-aid Data Gathering Algorithm. *Sensors* **2018**, *18*, 4110. [CrossRef]
41. Wei, D.; Wang, F.; Ma, H. Autonomous Path Planning of AUV in Large-Scale Complex Marine Environment Based on Swarm Hyper-Heuristic Algorithm. *Appl. Sci.* **2019**, *9*, 2654. [CrossRef]
42. Han, G.; Shen, S.; Wang, H.; Jiang, J.; Guizani, M. Prediction-Based Delay Optimization Data Collection Algorithm for Underwater Acoustic Sensor Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 6926–6936. [CrossRef]
43. Khan MT, R.; Ahmed, S.H.; Kim, D. AUV-aided Energy-efficient Clustering in the Internet of Underwater Things. *IEEE Trans. Green Commun. Netw.* **2019**, *3*, 1132–1141. [CrossRef]
44. Han, G.; Long, X.; Zhu, C.; Guizani, M.; Bi, Y.; Zhang, W. An AUV Location Prediction-based Data Collection Scheme for Underwater Wireless Sensor Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 6037–6049. [CrossRef]
45. Cai, S.; Zhu, Y.; Wang, T.; Xu, G.; Liu, A.; Liu, X. Data Collection in Underwater Sensor Networks based on Mobile Edge Computing. *IEEE Access* **2019**, *7*, 65357–65367. [CrossRef]
46. Nam, H. Data-Gathering Protocol based AUV Path-Planning for Long-Duration Cooperation in Underwater Acoustic Sensor Network. *IEEE Sens. J.* **2018**, *18*, 8902–8912. [CrossRef]
47. Huang, M.; Zhang, K.; Zeng, Z.; Wang, T.; Liu, Y. An AUV-assisted Data Gathering Scheme based on Clustering and Matrix Completion for Smart Ocean. *IEEE Internet Things J.* **2020**, *7*, 9904–9918. [CrossRef]
48. Veerachamy, R.; Ramar, R. Agricultural Irrigation Recommendation and Alert (AIRA) system using optimization and machine learning in Hadoop for sustainable agriculture. *Environ. Sci. Pollut. Res.* **2022**, *29*, 19955–19974. [CrossRef]