International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I12P123

Volume 72 Issue 12, 269-284, December 2024

Original Article

Octoroute: Revolutionizing Buoyant Sensor Mobility In Underwater Communication Networks

J. Divya Jose[1], D. Vimal Kumar[2]

[1]*Department of Computer Science, Nehru Arts and Science College, Tamil Nadu, India.*
[2]*Department of Computer Science, Rathinam College of Arts and Science, Tamil Nadu, India.*

[1]*Corresponding Author : divyajose369@gmail.com*

Received: 25 May 2024 Revised: 07 October 2024 Accepted: 24 October 2024 Published: 25 December 2024

Abstract - *In the environment of underwater communication networks, buoyant sensor mobility poses unique challenges that hinder efficient data transmission and network reliability. Traditional routing protocols struggle to navigate the dynamic underwater environment, leading to packet loss, high latency, and inefficient energy consumption. In response to these challenges, this paper introduces OctoRoute, a novel routing protocol designed to revolutionize buoyant sensor mobility in underwater communication networks. OctoRoute leverages Octopus Optimization (O^2) and Enhanced Greedy Perimeter Stateless Routing (EGPSR) techniques to dynamically adapt to changing environmental conditions, optimize routing paths, and maximize data transmission efficiency. Through comprehensive performance evaluations, OctoRoute consistently outperforms traditional protocols by achieving higher packet delivery ratios, lower packet drop ratios, increased throughput, reduced delay, and improved energy efficiency.*

Keywords - *OctoRoute, Buoyant sensor mobility, Underwater communication networks, Routing, Octopus optimization, Enhanced GPSR, Dynamic mobility.*

1. Introduction

Oil spills emerge as a formidable environmental threat, instigating far-reaching consequences on ecosystems and the diverse wildlife inhabiting them. The uncontrolled discharge of oil into aquatic environments initiates a cascade of long-lasting ecological damage, disrupting the intricate balance of marine life [1]. Underwater organisms, pivotal components of these ecosystems, bear the brunt of the impact as oil contamination permeates the water column, affecting their habitats and compromising their well-being [2]. The adverse consequences extend to coastal lands near the ocean, with the intricate interplay between terrestrial and marine environments amplifying the ecological repercussions. The formidable challenge lies in the immediate impact of oil spills and the complex difficulties associated with their timely detection and subsequent response [3]. To mitigate the profound consequences of oil spills, unravelling the nuances of these incidents and comprehending the intricacies inherent in their detection becomes paramount [2]. Buoyant Wireless Sensor Networks (B-WSN) are crucial guardians against oil spills, leveraging floating sensors on the water's surface. With the ability to monitor aquatic environments in real-time, B-WSN plays a pivotal role in preventing and mitigating oil spills and safeguarding ecosystems and marine life from potential environmental disasters[4]. B-WSN epitomizes a pioneering advancement in ecological monitoring, strategically positioning sensors to float gracefully on the water's surface. This paradigm shift introduces a dynamic approach to real-time data acquisition, particularly in critical applications such as oil spill detection, water quality monitoring, and ecosystem analysis[5]. The buoyant design of these sensors not only permits seamless adaptability to water movements but also poses unique challenges that necessitate innovative solutions in developing robust communication protocols, efficient routing strategies, and sustainable power management systems. By venturing into the intricacies of B-WSN, researchers and engineers aim to unlock new frontiers in surface-level wireless sensing, offering unparalleled insights into aquatic ecosystems [6]. In B-WSN, routing complexities unfold within the context of floating sensors on the water's surface. The limited operating range, influenced by acoustic or radio wave propagation in water, adds a layer of intricacy to routing protocols [7]. The highly mobile nature of buoyant nodes and unpredictable movement patterns necessitates routing algorithms capable of dynamically adapting to changing topologies. Efficient data aggregation and fusion are essential in managing communication costs and conserving energy in this distinctive network[8]. Routing protocols must seamlessly navigate mobility, communication constraints, and environmental variability in the buoyant sensor network, ensuring adequate data transmission and network longevity[9].

1.1. Problem Statement

The challenge of Buoyant Sensor Mobility within the context of oil spill monitoring presents a formidable obstacle in the realm of wireless sensor networks deployed on the water's surface. Developing effective routing algorithms tailored to the unpredictable movement patterns of buoyant sensors is imperative for ensuring efficient and timely detection of oil spills. Navigating water currents, tides, and environmental fluctuations, these sensors exhibit dynamic trajectories, necessitating innovative routing strategies capable of real-time adaptation. Traditional routing approaches prove inadequate in the face of such unpredictability. The problem underscores the urgency of creating adaptive solutions that enable seamless communication paths despite erratic movements and enhance the overall effectiveness of oil spill detection in Buoyant Wireless Sensor Networks in aquatic environments. Addressing this challenge is pivotal for unleashing the full potential of B-WSN in combating oil spills and fostering advancements in environmental monitoring, disaster response, and other applications reliant on robust wireless communication in dynamic aquatic settings.

1.2. Motivation

The motivation for addressing the challenge of Buoyant Sensor Mobility, particularly in the context of oil spill monitoring, emanates from the paramount importance of timely and effective detection in environmental preservation. The dynamic nature of aquatic environments necessitates the development of routing algorithms capable of adapting seamlessly to the unpredictable movement patterns exhibited by buoyant sensors on the water's surface. The urgency lies in mitigating the devastating consequences of oil spills, where swift detection is pivotal for initiating prompt response and containment measures. Overcoming the challenges posed by Buoyant Sensor Mobility paves the way for enhancing the capabilities of Buoyant Wireless Sensor Networks in real-time oil spill monitoring, ensuring the resilience of ecosystems and minimizing the ecological impact of such incidents. The motivation extends beyond oil spill scenarios, influencing a broader spectrum of applications reliant on robust wireless communication in dynamic aquatic settings, thereby contributing to advancements in environmental monitoring, disaster response, and scientific research.

1.3. Objective

This research aims to develop a bio-inspired routing protocol specifically tailored to address the challenge of Buoyant Sensor Mobility in the context of oil spill monitoring. The aim is to draw inspiration from collective behaviours observed in marine organisms to create an innovative routing solution. This bio-inspired protocol seeks to enhance the efficiency, reliability, and adaptability of B-WSN, focusing on optimizing communication paths among sensors for improved and timely oil spill detection. The research aims to contribute to advancements in environmental monitoring technologies, minimize ecological impact, and foster sustainable solutions for safeguarding aquatic ecosystems. The developed bio-inspired routing protocol may have broader applications in B-WSN operating in dynamic aquatic environments, impacting disaster response, ecological research, and environmental management.

2. Literature Review

"FuzzyClust"[10] proposes an effective data transmission strategy that combines energy-efficient clustering with a Fuzzy-Based Intrusion Detection System (IDS) routing approach. The system forms clusters of sensor nodes to streamline data transmission, reducing energy consumption. The fuzzy logic-based IDS enhances network security by detecting and mitigating potential threats. This dual approach ensures both energy efficiency and security in data transmission processes. "GenRoute"[11] introduces a cluster-based routing protocol designed for high scalability and low latency in time-sensitive wireless sensor networks. Utilizing genetic algorithms, the protocol efficiently organizes sensor nodes into clusters and optimizes routing paths. This ensures rapid data transmission and minimal delay, making it suitable for applications requiring timely data delivery, such as real-time environmental monitoring.

"HawkTrans"[12] offers an optimal data transmission solution for decentralized IoT and wireless sensor networks, utilizing Type-2 Fuzzy Harris Hawks Optimization. This approach combines the robustness of fuzzy logic with the adaptive capabilities of the Harris Hawks algorithm to optimize routing decisions. The result is efficient and reliable data transmission, enhancing the performance and longevity of decentralized networks. "AntLife"[13] focuses on extending the network lifetime of wireless sensor networks by implementing a modified Ant Colony Optimization (ACO) algorithm. This algorithm mimics the foraging behavior of ants to discover optimal routing paths, reducing energy consumption across the network. The modifications enhance the standard ACO algorithm, further improving its efficiency in maintaining network sustainability.

"AE-Leach"[14] introduces an incremental clustering approach to reduce energy consumption in wireless sensor networks. Building on the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol, AE-Leach incrementally forms clusters to balance energy use among sensor nodes. This approach minimizes redundant data transmission and extends the network's operational life by ensuring more efficient energy distribution. "MarineFusion"[15] presents a method for target localization in marine search and rescue operations using information fusion in wireless sensor networks. The system integrates data from multiple sensors to enhance the accuracy of target detection and positioning. This multi-sensor fusion approach improves the reliability of search and rescue missions, ensuring more precise and timely localization of targets in marine environments. "PowerQ"[16] presents a routing protocol that integrates power control with Q-learning for optical-acoustic hybrid underwater sensor networks. This protocol dynamically adjusts transmission power and optimizes routing paths based on learned experiences. Key contributions include combining power control with machine learning, improving energy efficiency, and routing reliability in hybrid communication environments. “AUVCollect”[17] introduces a power-efficient data collection scheme using Autonomous Underwater Vehicles (AUVs) for magnetic induction and acoustic hybrid networks. The scheme leverages AUVs to gather data from dispersed sensor nodes, optimizing energy use and ensuring reliable data transmission. Key contributions include using AUVs to enhance data collection efficiency and integrating hybrid communication methods for robust underwater connectivity.

“DepthSelect”[18] proposes a cluster-based routing protocol focusing on underwater sensor networks' depth source selection. This protocol organizes sensor nodes into clusters based on depth, optimizing data routing paths to minimize energy consumption and enhance communication reliability. Key contributions include the development of a depth-based clustering approach that improves routing efficiency and extends the network’s operational lifespan. “DeployModel”[19] presents an efficient deployment scheme coupled with network performance modeling for underwater sensor networks. This scheme focuses on optimal sensor node placement to maximize coverage and connectivity while minimizing deployment costs. Key contributions include integrating performance modeling to predict network behavior and formulating a deployment strategy that enhances overall network performance and reliability. “MARoute”[20] introduces a multi-agent reinforcement learning-based routing protocol that incorporates the concept of Value of Information (VoI) for underwater sensor networks. This protocol employs multiple agents to collaboratively learn and optimize routing paths, prioritizing data with higher informational value. Key contributions include applying multi-agent reinforcement learning to routing and using VoI to enhance data transmission efficiency, ensuring that critical information is delivered promptly.

“AUVLearn”[21] presents a data collection scheme for underwater sensor networks using Autonomous Underwater Vehicles (AUVs) and reinforcement learning. This scheme considers the influence of ocean currents on AUV navigation, optimizing data collection routes through learned experiences. Key contributions include integrating reinforcement learning to adapt to dynamic underwater conditions and using AUVs to improve data collection efficiency. “RouteReview”[22] comprehensively reviews routing protocols for underwater wireless sensor networks. It categorizes existing protocols into a detailed taxonomy and explores future research directions. Key contributions include a thorough analysis of current routing strategies, identifying gaps in the existing literature, and recommendations for future underwater sensor network routing developments. “CoverConnect”[23] introduces a deployment scheme for Autonomous Underwater Vehicles (AUVs) that ensures optimal coverage and connectivity. This scheme strategically positions AUVs to maximize the network’s coverage area and maintain robust communication links. Key contributions include the development of an AUV deployment strategy that enhances both coverage and connectivity, crucial for effective underwater monitoring and data collection.

“IoTRoute”[24] presents a novel routing method designed to increase efficiency in underwater wireless sensor networks by leveraging Internet of Things (IoT) principles. This method integrates IoT technologies to optimize routing paths and improve data transmission efficiency. Key contributions include the application of IoT concepts to underwater routing, providing a more efficient and scalable solution for underwater sensor networks. “RoboDeploy”[25] introduces a node deployment optimization strategy using intelligent algorithms and robotic collaboration. This approach employs advanced optimization techniques and coordinated robot deployment to position sensor nodes optimally. Key contributions include the combination of intelligent algorithms with robotic systems, enhancing deployment accuracy and network performance in underwater environments.

Bio inspired optimization is a concept of implementing the behavioral patterns of biologically living things from the real world into the digital world. The reason for implementing bio-inspired computing in computational tasks is to reduce complexity and simplify.“HOCOR”[26] introduces a hybrid optimization-based cooperative opportunistic routing protocol designed for underwater sensor networks. This protocol combines various optimization techniques to enhance routing decisions, ensuring efficient data transmission.

It leverages cooperative strategies among sensor nodes to dynamically adjust routing paths based on real-time network conditions. Key contributions include integrating hybrid optimization methods with cooperative routing, which significantly improves the network’s adaptability and performance in underwater environments. “MO-CBACORP”[27] proposes a multi-objective, cluster-based adaptive cognitive routing protocol designed for underwater monitoring wireless sensor networks. The protocol focuses on energy efficiency and security, utilizing cognitive techniques to route data while conserving energy adaptively. It enhances network resilience against potential threats and ensures efficient data transmission in underwater environments.

3. OctoRoute

The proposed protocol enhances the performance of the traditional EGPSR by bio inspired computing specifically inspired by the characteristics of the octopus.

3.1. Enhanced Greedy Perimeter Stateless Routing (EGPSR)

Greedy Perimeter Stateless Routing (GPSR) is a robust routing protocol employed in Wireless Sensor Networks (WSNs) to manage efficiently and direct data packets. Its primary objective within buoyant sensor mobility is to facilitate seamless communication and data transmission across a dynamic network topology without reliance on pre-established routing paths. By utilizing local information and leveraging geographic positioning, GPSR optimizes packet forwarding to conserve energy and ensure timely delivery.The purpose of GPSR in buoyant sensor mobility is to address the challenges posed by the unpredictable movement of sensors in aquatic environments. Traditional routing protocols may struggle to adapt to the dynamic nature of such environments, leading to increased latency and energy consumption. GPSR circumvents these issues by employing a greedy forwarding strategy, wherein each sensor node selects the next hop based on its proximity to the destination. This enables efficient routing even as sensors drift or reposition themselves underwater.

The Adam-Bashforth method can be integrated to enhance GPSR's effectiveness in buoyant sensor mobility scenarios. This numerical integr ation technique allows for predicting future sensor positions based on historical data, enabling proactive routing decisions to be made. By extrapolating sensor trajectories using velocity and acceleration information, GPSR can anticipate node movements and adjust routing paths accordingly, minimizing packet loss and optimizing network performance. By integrating the Adam-Bashforth method into GPSR for buoyant sensor mobility, the routing protocol becomes more proactive and adaptive, effectively addressing the challenges of dynamic underwater environments. This enhanced approach ensures efficient data transmission, minimizes energy consumption and enhances the overall resilience of the WSN.

3.1.1. Data Collection

GPSR starts with data collection to gather crucial information regarding the dynamic nature of sensor nodes in the buoyant environment. This step lays the foundation for subsequent routing decisions by providing essential input parameters such as position, velocity, and acceleration. The collected data predicts future sensor movements, which are vital for proactive routing strategies. Initially, let us denote the position of sensor node i at time t as shown in Equation (1).

$$Pos_i(t) = \big(x_i(t)\big), y_i(t), z_i(t)) \quad (1)$$

where$x_i(t), y_i(t)$, and $z_i(t)$ represent the Cartesian coordinates in the three-dimensional space. The position data is continuously updated as the sensor nodes move within the aquatic environment. To estimate the velocity of sensor node i at time t, we can employ numerical differentiation techniques such as finite differencing. Let Δt denote the time interval between successive position updates. Then, the velocity components $v_{x_i}(t), v_{y_i}(t)$, and $v_{z_i}(t)$ can be calculated is shown in Equations (2)-(4).

$$v_{x_i}(t) = \frac{x_i(t) - x_i(t - \Delta t)}{\Delta t} \quad (2)$$

$$v_{y_i}(t) = \frac{y_i(t) - y_i(t - \Delta t)}{\Delta t} \quad (3)$$

$$v_{z_i}(t) = \frac{z_i(t) - z_i(t - \Delta t)}{\Delta t} \quad (4)$$

These equations provide the instantaneous velocity of sensor node i along the three spatial dimensions based on position updates over a finite time interval.To estimate the acceleration of sensor node i at time t, we can further differentiate the velocity data obtained from the previous step. Let Δt remain at the same time interval. The acceleration components $a_{x_i}(t), a_{y_i}(t)$, and $a_{z_i}(t)$ can be mathematically represented in Equations (5)-(7).

$$a_{x_i}(t) = \frac{v_{x_i}(t) - v_{x_i}(t - \Delta t)}{\Delta t} \quad (5)$$

$$a_{y_i}(t) = \frac{v_{y_i}(t) - v_{y_i}(t - \Delta t)}{\Delta t} \quad (6)$$

$$a_{z_i}(t) = \frac{v_{z_i}(t) - v_{z_i}(t - \Delta t)}{\Delta t} \quad (7)$$

These equations yield the instantaneous acceleration of sensor node i along the three spatial dimensions based on velocity updates over a finite time interval.

3.1.2. Prediction

The prediction phase in enhanced GPSR plays a pivotal role in anticipating the future movements of sensor nodes within the buoyant environment. This step enables the protocol to proactively adjust routing paths and optimize packet forwarding strategies by extrapolating from historical data. Utilizing the collected position, velocity, and acceleration data, the Adam-Bashforth method can be employed to predict the future positions of sensor nodes. This numerical integration technique extrapolates the trajectory of each sensor node based on its current state and dynamics. Let Δt denote the time step for prediction. The position of sensor node i at time $t + \Delta t$can be estimated, mathematically represented in Equations (8)-(10).

$$Pos_i(t + \Delta t) = Pos_i(t) + v_{x_i}(t).\Delta t + \frac{1}{2} a_{x_i}(t).(\Delta t)^2 \quad (8)$$

$$Pos_i(t + \Delta t) = Pos_i(t) + v_{y_i}(t).\Delta t + \frac{1}{2} a_{y_i}(t).(\Delta t)^2 \quad (9)$$

$$Pos_i(t + \Delta t) = Pos_i(t) + v_{z_i}(t).\Delta t + \frac{1}{2} a_{z_i}(t).(\Delta t)^2 \quad (10)$$

These equations represent the predicted positions of sensor node i along the $x, y,$ and$z$ dimensions at time $t + \Delta t$, respectively. The first term accounts for the displacement based on current velocity, while the second term incorporates the effect of acceleration over the time interval Δt. Higher-order Adam-Bashforth methods can be employed to refine the accuracy of the prediction. For instance, the second-order Adam-Bashforth method incorporates information from two previous time steps to enhance prediction precision. The predicted position of sensor node i at time $t + \Delta t$ using the second-order method can be expressed as Equations (11)-(13).

$$Pos_i(t+\Delta t) = Pos_i(t) + v_{x_i}(t).\Delta t + \frac{1}{2}a_{x_i}(t).(\Delta t)^2 + \frac{5}{12}\left(a_{x_i}(t) - v_{x_i}(t-\Delta t)\right).(\Delta t)^3 \quad (11)$$

$$Pos_i(t+\Delta t) = Pos_i(t) + v_{y_i}(t).\Delta t + \frac{1}{2}a_{y_i}(t).(\Delta t)^2 \frac{5}{12}\left(a_{y_i}(t) - v_{y_i}(t-\Delta t)\right).(\Delta t)^3 \quad (12)$$

$$Pos_i(t+\Delta t) = Pos_i(t) + v_{z_i}(t).\Delta t + \frac{1}{2}a_{z_i}(t).(\Delta t)^2 + \frac{5}{12}\left(a_{z_i}(t) - v_{z_i}(t-\Delta t)\right).(\Delta t)^3 \quad (13)$$

These equations incorporate additional acceleration information from the previous time step, improving the accuracy of the predicted positions.

3.1.3. Routing Decision

Routing decisions connect the extrapolated future positions of sensor nodes to determine optimal routing paths. This phase facilitates proactive routing decisions aimed at minimizing latency and conserving energy within the wireless sensor network by considering the predicted locations of both source and destination nodes.

Initially let $Pos_s(t)$ and $Pos_d(t)$ denote the predicted positions of the source and destination nodes, respectively, at time t.These positions are obtained through the prediction phase using the Adam-Bashforth method. Then the distance $d_{sd}(t)$ between the predicted positions of the source and destination nodes can be calculated using the Euclidean distance formula, which is represented as Equation (14).

$$d_{sd}(t) = \sqrt{(x_s(t) - x_d(t))^2 + (y_s(t) - y_d(t))^2 + (z_s(t) - } \quad (14)$$

where$(x_s(t), y_s(t), z_s(t))$ and $(x_d(t), y_d(t), z_d(t))$ represent the predicted positions of the source and destination nodes, respectively, at time t.To determine the optimal next hop for packet forwarding, the routing decision step involves selecting neighbouring sensor nodes approaching the final destination node rather than the source node. This greedy forwarding strategy ensures progress towards the destination while exploiting local geographic information. Let $N_i(t)$ denote the set of neighbouring sensor nodes of sensor node i at time t. The routing decision can be mathematically expressed as Eq.(15).

$$N_{opt}(t) = \{n \in N_s(t) | d_{nd}(t) < d_{sd}(t)\} \quad (15)$$

where$N_{opt}(t)$ represents the set of optimal next-hop neighbours at time t, n is a neighbouring sensor node, and $d_{nd}(t)$ is the distance between the sensor node n and the $destination$ node at time t. To account for potential obstacles or boundary conditions in the network, a threshold distance δ can be introduced. Sensor nodes within this threshold distance of the destination node are considered candidate next-hop neighbours. The routing decision can be modified as shown in Equation (16).

$$N_{opt}(t) = \{n \in N_s(t) | d_{nd}(t) < d_{sd}(t) + \delta\} \quad (16)$$

This modification ensures robustness and adaptability in the face of changing network conditions.

3.1.4. Forwarding:

The forwarding phase in enhanced GPSR ensures efficient packet transmission from source to destination nodes within the wireless sensor network. Leveraging the optimal next-hop neighbours determined in the previous step, this phase involves greedily forwarding data packets toward the destination while considering the dynamic nature of the network topology.Initially let $N_{opt}(t)$ represent the optimal next-hop neighbours determined in the routing decision phase at time t. When a sensor node receives a data packet to be forwarded, it selects the nearest neighbour from the set $N_{opt}(t)$ as the next hop for packet transmission. Then $n_{min}(t)$ denote the sensor node from the set $N_{opt}(t)$ that is closest to the destination node at time t. The forwarding decision can be mathematically expressed in Equation (17).

$$n_{min}(t) = argmin_{n \in opt(t)} d_{nd}(t) \quad (17)$$

where$d_{nd}(t)$ represents the distance between sensor node n and the destination node at time t. Once the nearest neighbour $n_{min}(t)$ is determined, the data packet is forwarded to this neighbour for further transmission toward the destination node. The packet forwarding process continues iteratively until the packet reaches the destination node. To ensure robustness and adaptability in dynamic environments, the forwarding step may incorporate mechanisms for packet retransmission or route repair in case of packet loss or route failures. These mechanisms help maintain the reliability and resilience of data transmission within the wireless sensor network.

3.1.5. Adaptation

The Adaptation phase in enhanced GPSR is crucial for maintaining the efficiency and resilience of the routing protocol in dynamic wireless sensor networks. This phase involves continuously monitoring and updating the network state to adapt to changes in node mobility, topology, and environmental conditions. Let $d_{nd}(t)$ represent the distance between the current position of a sensor node and the destination node at time t. As the network topology evolves, this distance dynamically changes, impacting the routing decisions made by individual sensor nodes. To adapt to node mobility and topology changes, sensor nodes periodically update their position, velocity, and acceleration information based on new data collected from the environment. Let $Pos_i(t), v_i(t)$, and $a_i(t)$ denote the position, velocity, and acceleration of sensor node i at time t, respectively. The position update can be mathematically expressed in Equation (18).

$$Pos_i(t+\Delta t) = Pos_i(t) + v_i(t).\Delta t + \frac{1}{2}a_i(t).(\Delta t)^2 \tag{18}$$

where Δt *represents* the time interval between position updates, the velocity update equation can be shown in Equations (19).

$$a_i(t+\Delta t) = v_i(t) + a_i(t) + \Delta t \tag{19}$$

The acceleration *update* equation can be mathematically expressed as Equations (20).

$$a_i(t+\Delta t) = \frac{v_i(t+\Delta t) - v_i(t)}{\Delta t} \tag{20}$$

These equations allow sensor nodes to adapt to environmental changes by continuously updating their kinematic parameters. Sensor nodes may adjust their neighbour selection criteria based on the changing network topology to ensure efficient routing decisions. The adaptation step also encompasses mechanisms for handling network dynamics, such as node failures or link disruptions. In such cases, sensor nodes may initiate route repair procedures to establish alternative paths for data transmission. Route repair algorithms aim to restore network connectivity while minimizing disruption to ongoing communication.

3.1.6. Error Handling

Error handling in enhanced GPSR is essential for maintaining the reliability and robustness of the routing protocol in the face of unforeseen events such as packet loss, node failures, or link disruptions. This phase involves implementing mechanisms to detect, diagnose, and recover from errors to ensure uninterrupted data transmission within the wireless sensor network. Initially P_{loss} represent the probability of packet loss during transmission. Packet loss can occur due to various factors, such as signal interference, channel fading, or congestion. The probability of successful packet delivery $P_{success}$ can be expressed as:

$$P_{success} = 1 - P_{loss} \tag{21}$$

where $P_{success}$ represents the probability that a packet successfully reaches its destination. Sensor nodes may employ acknowledgement (ACK) mechanisms or timeout-based retransmission schemes to detect packet loss. Let $T_{timeout}$ denote the timeout period for packet retransmission. After the allotted time has elapsed, the sender will presume packet loss and begin retransmission if they have not received an acknowledgement. The probability of successful packet delivery can be affected by factors such as network congestion or node failures. To mitigate the impact of these factors, sensor nodes may implement congestion control mechanisms or route repair algorithms to establish alternative paths for data transmission. Sensor nodes may maintain backup routes or establish redundant links to handle node failures or link disruptions to ensure network connectivity. Let N_{backup} represent the set of backup routes available to a sensor node. In the event of a failure or disruption, the node can switch to a backup route for continued data transmission. Metrics like packet delivery ratio and end-to-end latency may be used to quantify the dependability of data transmission inside the wireless sensor network. The packet delivery ratio, abbreviated as PDR, is the percentage of transmitted packets that reach their destination without a hitch. The time a packet gets from its source to its destination node is called the end-to-end delay. This packet delivery ratio and delay are represented mathematically in Equations (22) and (23).

$$PDR = \frac{Number\ of\ successfully\ delivered\ packets}{Total\ number\ of\ packets\ sent} \tag{22}$$

$$Delay = \frac{Total\ transmission\ time\ of\ packet}{Number\ of\ hops} \tag{23}$$

These metrics provide insights into the reliability and performance of the routing protocol and help guide error-handling strategies.

3.2. Octopus Optimization:

Metaheuristic optimization method Octopus Optimization (O^2) takes its cues from how octopuses behave in the natural world. O^2 mimics the hunting behaviour of octopuses, which utilize their tentacles to explore and exploit their surroundings to capture prey efficiently. This algorithm is particularly effective for solving optimization problems, especially those with complex search spaces and multiple local optima.

3.2.1. Tentacle Movement

The Tentacle Movement is pivotal in exploring the search space and locating potential solutions. This step mimics octopus tentacles' flexible and adaptive movement, allowing them to navigate complex environments and search for prey efficiently. The movement of each tentacle can be represented using a stochastic process that incorporates both exploration and exploitation. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The tentacle's motion is represented in Equation (24).

$$P_i(t+\Delta t) = P_i(t) + \Delta P_i(t) \tag{24}$$

where$\Delta P_i(t)$ represents the incremental movement of tentacle i at time t. The variables, such as the tentacle's present location, the available local information, and random disturbances designed to promote exploration, come together to decide this gradual movement. To promote exploration, stochastic perturbations can be introduced to the movement of tentacles, simulating the unpredictable nature of environmental conditions. Let $\in_i(t)$ denote the random perturbation applied to tentacle i at time t. The incremental movement of the tentacle can be modified, as shown in Equation (25).

$$\Delta P_i(t) = \alpha . \nabla f(P_i(t)) + \beta . \epsilon_i(t) \qquad (25)$$

whereα and β are scaling factors that control the influence of the gradient of the objective function f and the stochastic perturbation, respectively. To ensure efficient search space exploration, tentacles may exhibit adaptive movement strategies that adjust their exploration based on the local characteristics of the optimization landscape. Let $\nabla f(P_i(t))$ denote the gradient of the objective function f at the current position of tentacle i. The movement of the tentacle can be guided by the gradient direction, which is mathematically represented in Equation (26).

$$\Delta P_i(t) = \alpha . \nabla f P_i(t)) \qquad (26)$$

This equation ensures that the tentacle moves toward the steepest ascent or descent of the objective function, facilitating efficient exploration and exploitation of promising regions of the search space. To prevent premature convergence and encourage diversity in exploration, tentacles may incorporate mechanisms for adaptive step size adjustment. Let $\eta_i(t)$ denote the step size of tentacle i at time t. The movement of the tentacle with adaptive step size adjustment can be mathematically represented as Equation (27).

$$P_i(t + \Delta t) = P_i(t) + \eta_i(t). \Delta P_i(t)) \qquad (27)$$

where$\eta_i(t)$ is dynamically adjusted based on the convergence status and the quality of solutions encountered, ensuring a balance between exploration and exploitation.

3.2.2. Local Search

In this phase, each tentacle performs a local search around its current position to exploit promising regions of the search space. This step aims to refine the solutions obtained during the exploration phase and improve their quality by focusing on local neighbourhoods. The local search can be mathematically formulated, let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. The local search aims to find a nearby position that improves the objective function value. One common strategy for local search is gradient descent, where the tentacle iteratively moves toward the direction of the steepest descent of the objective function. Let$\Delta f(P_i(t))$ denote the gradient of the objective function at the current position of tentacle i. The gradient direction that can guide the tentacle's movement in the local search phase is mathematically represented as Equation (28).

$$P_i(t + \Delta t) = P_i(t) - \gamma . \Delta f(P_i(t)) + \beta . \epsilon_i(t) \qquad (28)$$

whereβ controls the influence of the stochastic perturbation on the movement of the tentacle. Furthermore, tentacles may adjust their step size dynamically to ensure efficient exploration of the local neighbourhood based on the convergence status and the quality of solutions encountered. Let $\eta_i(t)$ denote the step size of tentacle i at time t. The movement of the tentacle with adaptive step size adjustment can be represented as Equation (29).

$$P_i(t + \Delta t) = P_i(t) - \eta_i(t). \Delta f(P_i(t)) \qquad (29)$$

where$\eta_i(t)$ is dynamically adjusted based on the convergence status and the quality of solutions encountered during the local search.

3.2.3. Global Exploration

Tentacles collaborate to share information and collectively explore diverse regions of the search space to prevent premature convergence and discover high-quality solutions. Global exploration involves leveraging the collective intelligence of tentacles to guide the search towards promising regions of the search space. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. One common strategy for global exploration is to promote diversity in the search by encouraging tentacles to explore regions of the search space that have not been adequately explored. This can be achieved through mechanisms such as random perturbation or adaptive exploration. Let $\epsilon_i(t)$denote the random perturbation applied to tentacle i at time t. The movement of the tentacle with stochastic perturbation for global exploration can be represented mathematically in Equation (30).

$$P_i(t + \Delta t) = P_i(t) + \beta . \epsilon_i(t) \qquad (30)$$

whereβ controls the influence of the stochastic perturbation on the movement of the tentacle. To ensure that tentacles explore diverse regions of the search space, global exploration may incorporate mechanisms for information sharing and collaboration among tentacles. Tentacles may exchange information about their experiences and solutions encountered during the search to guide the exploration towards promising regions. Let $P_{best}(t)$denote the best solution encountered by tentacles up to time t. Tentacles may adjust their movement based on the distance between their current position and the best solution encountered. The movement of the tentacle towards the best solution can be represented as Equation (31).

$$P_i(t + \Delta t) = P_i(t) + \gamma . \left(P_{best}(t) - P_i(t)\right) \qquad (31)$$

whereγ controls the magnitude of the movement towards the best solution. To prevent premature convergence and encourage exploration, tentacles may adjust their movement dynamically based on the convergence status and the quality of solutions encountered. The movement of the tentacle with adaptive step size adjustment for global exploration can be represented mathematically in Equation (32).

$$P_i(t+\Delta t) = P_i(t) + \eta_i(t).\left(P_{best}(t) - P_i(t)\right) \qquad (32)$$

where$\eta_i(t)$ is dynamically adjusted based on the convergence status and the quality of solutions encountered during the search.

3.2.4. Dynamic Adjustment

Dynamic adjustment dynamically adjusts their movement strategies based on the convergence status and the quality of solutions encountered during the search. This dynamic adjustment ensures adaptability and responsiveness to changes in the optimization landscape. It involves modifying the movement parameters of tentacles in real time to balance exploration and exploitation effectively. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. One common strategy for dynamic adjustment is to adapt the tentacles' step size based on the search's convergence status. Let $\eta_i(t)$denote the step size of tentacle i at time t. The step size can be dynamically adjusted using a feedback mechanism that evaluates the progress of the search. The movement of the tentacle with adaptive step size adjustment is represented in Equation (33).

$$P_i(t+\Delta t) = P_i(t) + \eta_i(t).\Delta P_i(t) \qquad (33)$$

where$\Delta P_i(t)$ represents the incremental movement of tentacle i at time t. To ensure efficient exploration and exploitation, tentacles may adjust their movement based on the quality of solutions encountered during the search. Let $f_{best}(t)$denote the best objective function value encountered by tentacles up to time t.

Tentacles may adjust their movement towards regions of the search space that show improvement over time. The movement of the tentacle towards the best solution is represented mathematically in Equation (34).

$$P_i(t+\Delta t) = P_i(t) + \gamma.\left(f_{best}(t) - f(P_i(t))\right) \qquad (34)$$

whereγ controls the magnitude of the movement towards regions of improvement. To prevent stagnation and encourage exploration, tentacles may incorporate mechanisms for random perturbation or diversification of movement.

Let $\epsilon_i(t)$ denote the random perturbation applied to tentacle $i1$ at time t. The movement of the tentacle with stochastic perturbation for dynamic adjustment can be depicted in Equation (30).Whereβ controls the influence of the stochastic perturbation on the movement of the tentacle.

3.2.5. Flexible Exploration

This phase exhibits flexibility in exploration, allowing tentacles to adapt their movements based on the optimisation problem's characteristics and the search's convergence status. The flexible exploration involves incorporating diverse movement strategies that adapt to the local landscape of the search space. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. One strategy for flexible exploration is to dynamically adjust the movement parameters of tentacles based on the characteristics of the optimization problem. Let α denote a parameter that controls the influence of a particular movement strategy. Tentacles may switch between different movement strategies based on the characteristics of the optimization landscape. The movement of the tentacle with flexible exploration can be represented as Equation (35).

$$P_i(t+\Delta t) = P_i(t) + \alpha.\Delta P_i(t) \qquad (35)$$

where$\Delta P_i(t)$ represents the incremental movement of tentacle i at time t based on a particular movement strategy. To promote efficient exploration, tentacles may incorporate mechanisms for adaptive step size adjustment that vary based on the local curvature of the optimization landscape. Let $\eta_i(t)$denote the step size of tentacle i at time t. The step size can be dynamically adjusted based on local gradient information. The movement of the tentacle with adaptive step size adjustment for flexible exploration can be represented mathematically in Equation (36).

$$P_i(t+\Delta t) = P_i(t) + \eta_i(t).\nabla f(P_i(t)) \qquad (36)$$

where$\nabla f(P_i(t))$ represents the gradient of the objective function at the current position of tentacle i. To encourage exploration in regions of the search space that have not been adequately explored, tentacles may incorporate mechanisms for random perturbation or diversification of movement. Let $\epsilon_i(t)$ denote the random perturbation applied to tentacle i at time t as expressed mathematically in Equation (30), $\epsilon_i(t)$ represents a random perturbation applied to the movement of tentacle i.

3.2.6. Adaptive Strategy

In this phase, the algorithm adapts its search strategy based on the convergence status and the quality of solutions encountered during the search. This adaptive strategy ensures responsiveness to changes in the optimization landscape and promotes efficient convergence towards high-quality solutions. The adaptive strategy involves dynamically adjusting the movement parameters and exploration strategies of tentacles based on the convergence status of the search. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space.One strategy for adaptive strategy is to adjust the movement parameters of tentacles based on the convergence status of the search. Let

$\gamma(t)$ denote a parameter that controls the magnitude of the movement towards regions of improvement. Tentacles may increase their movement towards regions of improvement as the search progresses. The movement of the tentacle with an adaptive strategy can be mathematically represented as Equation (37).

$$P_i(t+\Delta t) = P_i(t) + \gamma(t).\left(f_{best}(t) - f(P_i(t))\right) \quad (37)$$

where $f_{best}(t)$ represents the best objective function value encountered by tentacles up to time t. To prevent premature convergence and encourage exploration, tentacles may adjust their movement strategies based on the quality of solutions encountered during the search. Let $\alpha(t)$ denote a parameter that controls the influence of a particular movement strategy. Tentacles may switch between different movement strategies based on the convergence status of the search. The movement of the tentacle with an adaptive strategy is shown in Equation (38).

$$P_i(t+\Delta t) = P_i(t) + \alpha(t).\Delta P_i(t) \quad (38)$$

where $\Delta P_i(t)$ represents the incremental movement of tentacle i at time t based on a particular movement strategy. To promote efficient exploration, tentacles may incorporate mechanisms for adaptive step size adjustment that vary based on the convergence status of the search. Let $\eta_i(t)$ denote the step size of tentacle i at time t. The step size can be dynamically adjusted based on the convergence status of the search. The movement of the tentacle with adaptive step size adjustment for adaptive strategy can be shown in Equation (39).

$$P_i(t+\Delta t) = P_i(t) + \eta_i(t).\Delta f(P_i(t)) \quad (39)$$

where $\Delta f(P_i(t))$ represents the gradient of the objective function at the current position of tentacle i.

3.2.7. Convergence Mechanism

Convergence Mechanism focuses on fine-tuning the convergence behaviour of the algorithm to ensure it efficiently converges towards high-quality solutions while preventing premature convergence. It involves incorporating mechanisms to balance exploration and exploitation to guide the search for optimal solutions. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. One common strategy for the convergence mechanism is to adaptively adjust the movement parameters and exploration strategies of tentacles based on the convergence status of the search. Let $\beta(t)$ denote a parameter that controls the influence of the convergence mechanism. Tentacles may increase their movement towards regions of improvement as the search progresses while ensuring sufficient exploration to prevent premature convergence. The movement of the tentacle with a convergence mechanism can be mathematically represented in Equation (40).

$$\begin{aligned} P_i(t+\Delta t) = P_i(t) + \beta(t)\left(f_{best}(t) - f\left(P_i(t)\right)\right) \\ + \left(1-\beta(t)\right).\Delta P_i(t) \end{aligned} \quad (40)$$

where $f_{best}(t)$ represents the best objective function value encountered by tentacles up to time t, and $\Delta P_i(t)$ represents the incremental movement of tentacle i at time t based on a particular movement strategy. To promote efficient convergence, tentacles may incorporate mechanisms for adaptive step size adjustment that vary based on the convergence status of the search. Let $\gamma(t)$ denote a parameter that controls the magnitude of the movement towards regions of improvement. The step size can be dynamically adjusted based on the convergence status of the search. The movement of the tentacle with convergence mechanism and adaptive step size adjustment can be represented in Equation (41).

$$\begin{aligned} P_i(t+\Delta t) = P_i(t) \\ + \gamma(t)\left(f_{best}(t)\right. \\ \left. - f\left(P_i(t)\right)\right).\nabla f\left(P_i(t)\right) \end{aligned} \quad (41)$$

where $\nabla f\left(P_i(t)\right)$ $r$represents the gradient of the objective function at the current position of tentacle i. To prevent stagnation and encourage exploration, tentacles may incorporate mechanisms for random perturbation or diversification of movement. Let $\epsilon_i(t)$ denote the random perturbation applied to tentacle i at time t. The movement of the tentacle with stochastic perturbation for the convergence mechanism is represented mathematically in Equation (30), where $\epsilon_i(t)$ represents a random perturbation applied to the movement of tentacle i.

3.2.8. Efficient Convergence

In this phase, the algorithm focuses on further improving the convergence speed and efficiency towards high-quality solutions while ensuring robustness and stability throughout the optimization process. Efficient convergence involves fine-tuning the convergence behaviour of the algorithm by dynamically adjusting the movement parameters, exploration strategies, and step sizes of tentacles based on the convergence status and the quality of solutions encountered during the search. Let $P_i(t)$ denote the position of tentacle i in the search space at time t. The objective function f evaluates the quality of a solution at a given position in the search space. One strategy for efficient convergence is to incorporate mechanisms for dynamic step size adjustment that adapt based on the convergence status of the search. Let $\delta(t)$ denote a parameter that controls the magnitude of the movement towards regions of improvement. The step size can be dynamically adjusted based on the convergence status of the search. The movement of the tentacle with efficient convergence and adaptive step size adjustment is shown in Equation (42).

$$\begin{aligned} P_i(t+\Delta t) = P_i(t) \\ + \delta(t).\left(f_{best}(t)\right. \\ \left. - f\left(P_i(t)\right)\right).\nabla f\left(P_i(t)\right) \end{aligned} \quad (42)$$

where f_{best} represents the best objective function value encountered by tentacles up to time t, and $\nabla f\left(P_i(t)\right)$ represents the gradient of the objective function at the current position of tentacle i. To prevent stagnation and encourage exploration, tentacles may incorporate mechanisms for random perturbation or diversification of movement. Let $\zeta_i(t)$ denote the random perturbation applied to tentacle i at time t. The movement of the tentacle with stochastic perturbation for efficient convergence is depicted in Equation (43).

$$P_i(t+\Delta t) = P_i(t) + \zeta_i(t) \tag{43}$$

where $\zeta_i(t)$ represents a random perturbation applied to the movement of tentacle i. To promote efficient convergence towards high-quality solutions, tentacles may dynamically adjust their movement strategies based on the search's convergence status and the optimisation landscape's characteristics. The movement of the tentacle with an efficient convergence strategy is mathematically represented in Equation (44).

$$P_i(t+\Delta t) = P_i(t) + \omega(t).\left(f_{best}(t) - f(P_i(t))\right) + (1-\omega(t)).\Delta P_i(t) \tag{44}$$

where $\omega(t)$ represents a parameter that controls the influence of the efficient convergence strategy on the movement of tentacle i.

3.3. OctoRoute Formation

OctoRoute is a fusion of O^2 with EGPSR tailored for B-WSNs. A dynamic routing protocol orchestrates the movement of tentacle-like nodes to optimize data routing in B-WSNs. It synergizes O^2's adaptability with EGPSR's efficiency in navigating the dynamic underwater environment. This amalgamation achieves robust and efficient data delivery while accommodating buoyant sensor mobility. OctoRoute combines the resilience and adaptability of O^2 with the efficiency and scalability of EGPSR, providing a versatile solution for buoyant WSNs. This fusion enables seamless communication in dynamic underwater environments, supporting various applications such as environmental monitoring, underwater exploration, and marine research.

3.3.1. Tentacle Formation

Tentacle Formation establishes a dynamic network topology resembling tentacles in B-WSNs. This phase orchestrates the deployment and coordination of nodes to create a flexible and adaptive infrastructure. Tentacle formation involves positioning nodes underwater to maximize network coverage and connectivity while minimizing energy consumption. Let $P_i(t)$ represent the position of node i in the three-dimensional Cartesian coordinate system at time t. The objective is to distribute nodes evenly across the underwater space to ensure efficient communication and data relay. One approach to tentacle formation is to employ a distributed deployment strategy that leverages spatial distribution algorithms. Let D_i denote the desired spatial distribution of node i within the underwater environment. The movement of nodes towards their desired positions can be depicted in Equation (45).

$$P_i(t+\Delta t) = P_i(t) + \alpha.\left(D_i - P_i(t)\right) \tag{45}$$

where α controls the rate of movement towards the desired spatial distribution. To optimize network coverage and connectivity, nodes may adjust their positions based on local environmental factors and neighbouring node positions. Let $N_i(t)$ represent the set of neighbouring nodes of node i at time t. The movement of nodes towards optimal positions considering local information can be represented mathematically in Equation (46).

$$P_i(t+\Delta t) = P_i(t) + \sum\nolimits_{j\in N_i(t)} \beta_{ij}.\left(P_j(t) - P_i(t)\right) \tag{46}$$

where β_{ij} represents the influence of neighbouring node j on the movement of node i. To ensure robustness and adaptability, nodes may adjust their positions dynamically based on changes in environmental conditions. Let $E_i(t)$ denote the environmental parameters affecting node i at time t. The movement of nodes with dynamic environmental adaptation can be shown in Equation (47).

$$P_i(t+\Delta t) = P_i(t) + \gamma.\frac{\partial E_i(t)}{\partial t} \tag{47}$$

where γ controls the magnitude of adjustment based on the rate of change in environmental parameters.

3.3.2. Local Exploration

Expanding from the tentacle formation phase, the second step in OctoRoute is "Local Exploration." In this phase, nodes conduct localized searches to identify nearby nodes and assess their connectivity and proximity. This localized exploration facilitates the establishment of efficient communication links and enables nodes to gather information about their immediate surroundings. Mathematically, local exploration involves nodes dynamically adjusting their positions based on local environmental conditions and the positions of neighbouring nodes. Let $P_i(t)$ denote the position of node i in the three-dimensional Cartesian coordinate system at the time t, and $N_i(t)$ represent the set of neighbouring nodes of node i at time t. One strategy for local exploration is to employ a gradient descent approach to move nodes towards regions of higher connectivity or information density. Let $C_i(t)$ represent the connectivity or information density at node i at time t. The movement of node i towards regions of higher connectivity or information density can be mathematically represented in Equation (48).

$$P_i(t+\Delta t) = P_i(t) + \alpha.\nabla C_i(t) \tag{48}$$

where $\nabla C_i(t)$ represents the gradient of connectivity or information density at node i at time t, and α controls the rate of movement towards regions of higher connectivity or information density. To optimize local exploration, nodes may adjust their positions based on the positions and movements of neighbouring nodes. Let $V_{ij}(t)$represent the relative velocity between nodes i and j at time t. The movement of a node i considering the relative velocities of neighboring nodes can be shown in Equation (49).

$$P_i(t+\Delta t) = P_i(t) + \sum\nolimits_{j \in N_i(t)} \beta_{ij}.V_{ij}(t) \qquad (49)$$

whereβ_{ij} represents the influence of neighbouring node j on the movement of node i. To ensure adaptability and responsiveness to changes in local environmental conditions, nodes may incorporate mechanisms for dynamic adjustment of movement parameters. Let $E_i(t)$denote the environmental parameters affecting node i at time t. The movement of node i with dynamic environmental adaptation can be mathematically represented in Equation (50).

$$P_i(t+\Delta t) = \sum_{j=0}^{N-1} P_i(t) + \gamma.\frac{\partial E_i(t)}{\partial t} \qquad (50)$$

whereγ controls the magnitude of adjustment based on the rate of change in environmental parameters.

3.3.3. Global Pathfinding

In this phase, nodes collaborate to explore diverse paths and collectively identify the optimal route towards the destination. This global pathfinding strategy enables efficient data transmission and ensures robustness in communication links. It involves nodes exchanging information and collectively determining the optimal route towards the destination. Let$P_i(t)$denote the position of node i in the three-dimensional Cartesian coordinate system at time t, and D represents the destination node. One approach to global pathfinding is to employ a distributed routing algorithm that utilizes information exchange among neighbouring nodes to propagate routing information. Let $R_i(t)$denote the routing information available at node i at time t. The propagation of routing information among neighbouring nodes is represented mathematically in Equation (51).

$$R_i(t+\Delta t) = \sum\nolimits_{j \in N_i(t)} \beta_{ij}.R_j(t) \qquad (51)$$

whereβ_{ij} represents the influence of neighbouring node j on the routing information at node i. Additionally, nodes may exchange information about the quality of communication links and the distance to the destination to determine the optimal route towards the destination. Let $Q_i(t)$ represent the quality of communication link at node i at time t, and $D_i(t)$ denote the distance to the destination from node i at time t. The optimal route computation considering link quality and distance to the destination can be depicted in Equation (52).

$$P_i(t+\Delta t) = P_i(t) - \alpha.Q_i(t) - \gamma.D_i(t) \qquad (52)$$

whereα and γ control the influence of link quality and distance to the destination, respectively.

Furthermore, to ensure robustness in communication links and adaptability to changes in the network topology, nodes may incorporate mechanisms for dynamic adjustment of routing paths. Let $E_i(t)$denote the environmental parameters affecting the node

$$P_i(t+\Delta t) = P_i(t) - \delta.\frac{\partial E_i(t)}{\partial t} \qquad (53)$$

where in Equation (53), δ controls the magnitude of adjustment based on the rate of change in environmental parameters.

3.3.4. Adaptive Navigation

Nodes dynamically adjust their movements based on environmental factors, such as current velocity and water depth, to optimize path traversal. This adaptive navigation strategy enhances the efficiency and robustness of data transmission in B-WSNs. Adaptive navigation involves nodes dynamically adjusting their positions and movement parameters based on environmental conditions.

Let $P_i(t)$ denote the position of node i in the three-dimensional Cartesian coordinate system at time t, and $V_i(t)$represent the velocity of node i at time t. One strategy for adaptive navigation is to incorporate mechanisms for adjusting movement parameters based on the current velocity of nodes. Let $V_{curr}(t)$denote the current velocity of nodes at time t. The adjustment of movement parameters based on current velocity can be represented mathematically in Equation (54).

$$V_i(t+\Delta t) = V_i(t) - \alpha.(V_{curr}(t) - V_i(t)) \qquad (54)$$

whereα controls the rate of adjustment based on the difference between the current velocity and the velocity of node i. To optimize path traversal in varying water depths, nodes may adjust their positions and movement parameters based on the water depth encountered along the route. Let $D_i(t)$denote the water depth encountered by node i at time t. Adjusting positions and movement parameters based on water depth can be mathematically represented as Eq.(55) and Equation (56).

$$P_i(t+\Delta t) = P_i(t) + \beta.\nabla D_i(t) \qquad (55)$$

$$V_i(t+\Delta t) = V_i(t) + \beta.\nabla D_i(t) \qquad (56)$$

whereβ and γ control the magnitude of adjustment based on the water depth gradient encountered by node i. To ensure adaptability to changing environmental conditions, nodes may incorporate mechanisms for dynamic adjustment of movement parameters based on environmental parameters. Let $E_i(t)$denote the environmental parameters affecting node i at time t. The adjustment of movement parameters based on environmental conditions can be depicted in Equation (57).

$$V_i(t+\Delta t) = V_i(t) + \delta.\frac{\partial E_i(t)}{\partial t} \qquad (57)$$

whereδ controls the magnitude of adjustment based on the rate of change in environmental parameters.

3.3.5. Efficient Data Transmission

In Efficient Data Transmission, the nodes utilize optimized routes to relay data packets efficiently, minimizing latency and maximizing throughput. This efficient data transmission approach improves the communication network's overall performance and reliability in buoyant WSNs. It involves nodes dynamically adjusting their positions and movement parameters to optimize the transmission of data packets. Let $P_i(t)$denote the position of node i in the three-dimensional Cartesian coordinate system at time t, and $R_i(t)$ represent the routing information available at node i at time t. One strategy for efficient data transmission is to incorporate mechanisms for selecting the optimal route based on the available routing information. Let D represent the destination node. The optimal route selection towards the destination can be mathematically represented in Equation (58).

$$OptimalRoute_i(t) = argmin_{j\in R_i(t)} Distance\left(P_i(t), P_j(t)\right) \quad (58)$$

where$Distance\left(P_i(t), P_j(t)\right)$ represents the distance between node i and node j at time t. To minimize latency and maximize throughput, nodes may adjust their transmission rates based on the quality of communication links and the distance to the destination. Let $Q_i(t)$represent the quality of communication link at node i at time t. The adjustment of transmission rates based on link quality is shown in Equation (59).

$$TransmissionRate_i(t) = \frac{1}{Q_i(t)} \quad (59)$$

To ensure reliable data transmission, nodes may incorporate error detection and correction mechanisms. Let $E_i(t)$denote the error rate at node i at time t. The adjustment of transmission parameters based on error rate is shown in Equation (60).

$$Efficiency_i(t) = 1 - E_i(t) \quad (60)$$

3.3.6. Resilient Adaptation

In Resilient Adaptation, nodes continuously adapt to changes in the underwater environment, ensuring robust and reliable data delivery despite fluctuations in network conditions. Resilient adaptation enhances the overall resilience and stability of the communication network in buoyant WSNs. It involves nodes dynamically adjusting their positions, movement parameters, and transmission strategies based on changes in environmental conditions. Let $P_i(t)$denote the position of node i in the three-dimensional Cartesian coordinate system at time t, and $E_i(t)$represent the environmental parameters affecting node i at time t. Incorporate mechanisms for dynamic adjustment of movement parameters based on changes in environmental conditions. Let $\frac{\partial E_i(t)}{\partial t}$ denote the rate of change in environmental parameters affecting node i at time t. Adjusting movement parameters based on environmental conditions can be represented mathematically in Equation (57).whereδ controls the magnitude of adjustment based on the rate of change in environmental parameters. To ensure robust and reliable data delivery, nodes may adapt their transmission strategies based on changes in network conditions. Let $Q_i(t)$ represent the quality of communication link at node i at time t, and E_{max}denote the maximum allowable error rate. The adjustment of transmission strategies based on changes in link quality can be mathematically represented in Equation (61).

$$TransmissionStrategyV_i(t) = \begin{cases} High, & if\ Q_i(t) > Q_{threshold} \\ Low, & otherise \end{cases} \quad (61)$$

where$Q_{threshold}$ represents the minimum acceptable link quality for a high transmission strategy, η represents the efficiency factor in case of node fault.

3.5. Advantages of OctoRoute

OctoRoute, the fusion of O^2 with EGPSR for B-WSNs, offers several advantages that make it a promising solution for underwater communication.

- Optimized Routing: OctoRoute leverages the adaptability and efficiency of Octopus Optimization to establish optimized routing paths in B-WSNs. By dynamically adjusting node movements and transmission strategies, OctoRoute optimizes path traversal, minimizing latency and maximizing throughput. This optimized routing ensures efficient data transmission and enhances the overall performance of the communication network.
- Flexible Network Topology: One of the significant advantages of OctoRoute is its ability to establish a flexible network topology resembling tentacles in B-WSNs. The distributed deployment strategy employed in tentacle formation enables nodes to position themselves adaptively, ensuring comprehensive coverage and connectivity throughout the underwater environment. This flexible network topology facilitates robust communication links and enables effective data relay in dynamic underwater scenarios.
- Dynamic Adaptation: OctoRoute excels in dynamic adaptation to changes in the underwater environment. Through resilient adaptation mechanisms, nodes continuously adjust their positions, movement parameters, and transmission strategies based on environmental conditions. This dynamic adaptation ensures robust and reliable data delivery despite fluctuations in network conditions, enhancing the overall resilience and stability of the communication network.
- Efficient Data Transmission: Efficient data transmission is another key advantage of OctoRoute. By selecting optimal routes, adjusting transmission rates based on link quality, and incorporating mechanisms for error detection

and correction, OctoRoute optimizes data transmission efficiency. This efficient data transmission minimizes latency, maximizes throughput, and ensures timely delivery of critical information in buoyant WSNs.

- Adaptive Navigation: OctoRoute's adaptive navigation strategy enables nodes to navigate efficiently underwater. Nodes optimise path traversal and ensure efficient communication links by dynamically adjusting their positions and movement parameters based on environmental factors such as current velocity and water depth. By improving the communication network's overall performance and reliability, this adaptive navigation makes data transfer in dynamic underwater circumstances smooth.
- Resilient Communication: OctoRoute enhances the resilience of communication in buoyant WSNs through its robust fault tolerance mechanisms. By incorporating fault detection and recovery strategies, nodes can quickly recover from communication failures and maintain uninterrupted data transmission. This resilient communication ensures continuous operation of the network, even in challenging underwater conditions, enhancing the reliability and availability of the communication infrastructure.

4. Simulation Settings and Parameters

Simulation involves creating a model or representation of a real-world process or system to study its behavior under different conditions. It allows for experimentation and analysis without the need for physical prototypes. NS-3 is a discrete-event network simulator primarily used for research and educational purposes.

Table 1. Simulation Settings and Parameters

Parameter	Setting / Values
Simulator	ns-3
Simulation Time	1000 seconds
Network Topology	3D Grid
Number of Nodes	100
Node Placement	Random (buoyant movement)
Mobility Model	Random Waypoint
Routing Protocol	GPSR (Greedy Perimeter Stateless Routing)
Channel Model	Two-Ray Ground Reflection
Transmission Range	1250 meters
Packet Size	64 bytes
Traffic Type	CBR (Constant Bit Rate)
Application Layer	UDP
Propagation Delay	Uniform Random Variable(0.01)
MAC Layer	IEEE 802.15.4
Energy Model	Basic Energy Model
Simulation Output	Trace files and pcap files

It provides a comprehensive set of tools for simulating various types of networks, including WSNs. NS-3 supports different routing protocols, mobility models, and network topologies, making it a versatile platform for simulating complex network scenarios. The settings outlined in this table provide a comprehensive framework for simulating buoyant WSNs using the GPSR protocol in ns-3.

5. Results and Discussions

The packet delivery ratio (PDR) was evaluated across node densities using three routing protocols: MO-CBACORP, HOCOR, and OctoRoute. As depicted in Figure 1, OctoRoute consistently outperformed MO-CBACORP and HOCOR across all node densities. At 50 nodes, OctoRoute achieved a PDR of 74.36%, compared to MO-CBACORP (56.55%) and HOCOR (62.91%). This trend persisted as node density increased, with OctoRoute maintaining the highest PDR. On average, OctoRoute demonstrated the highest PDR at 68.253%, followed by HOCOR (54.456%) and MO-CBACORP (45.498%).

The results suggest that OctoRoute is more robust in delivering packets across varying node densities than the other protocols, making it a promising choice for wireless network routing. The Packet Drop Ratio (PDPR) is another crucial metric for assessing the performance of routing protocols in B-WSNs. PDPR represents the percentage of packets that fail to reach their intended destinations due to various network factors. As illustrated in Figure 1, OctoRoute consistently exhibits the lowest packet drop ratio across different node densities compared to MO-CBACORP and HOCOR. At a node density of 500, OctoRoute achieves the lowest PDR of 37.74%, followed by HOCOR (54.16%) and MO-CBACORP (66.96%). On average, OctoRoute maintains the lowest PDPR of 31.747%, demonstrating its robustness in packet delivery under buoyant WSN conditions. These results highlight OctoRoute as the preferred routing protocol for minimizing packet loss and ensuring reliable data transmission in underwater sensor networks. Selecting OctoRoute can significantly enhance the network's performance and reliability in real-world applications.

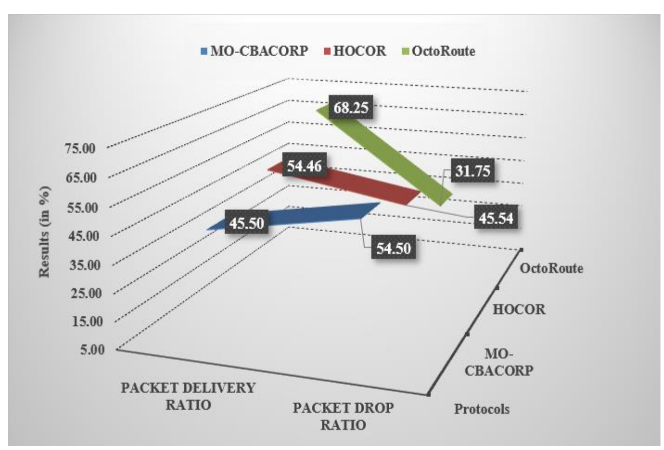

Fig. 1 Packet Delivery Ratio and Packet Drop Ratio

Throughput refers to the rate of successful data transmission over a communication channel within a specified period, typically measured in bits per second (bps) or packets per second (pps). In the context of B-WSNs, throughput indicates the network's capacity to deliver data packets efficiently under varying node densities. As portrayed in Figure 2, OctoRoute consistently demonstrates higher throughput than MO-CBACORP and HOCOR across different node densities. At a node density of 500, OctoRoute achieves the highest throughput of 72.48 kbps, outperforming MO-CBACORP (43.79 kbps) and HOCOR (56.34 kbps). On average, OctoRoute sustains a superior throughput of 64.457 kbps, which facilitates faster and more reliable data transmission in B-WSN environments. These results emphasize OctoRoute as a preferred routing protocol for maximizing network efficiency and ensuring timely data delivery in underwater sensor networks, enhancing overall system performance. Delay in a network context refers to when a packet travels from its source to its destination. It encompasses various factors such as processing, queuing, transmission, and propagation delays. In B-WSNs, delay is a critical metric affecting the responsiveness and efficiency of data transmission. As shown in Figure 3, OctoRoute consistently exhibits lower delay than MO-CBACORP and HOCOR across different node densities. At a node density of 500, OctoRoute achieves the lowest delay of 9521.25 ms, followed by HOCOR (12646 ms) and MO-CBACORP (13916 ms). On average, OctoRoute maintains the lowest delay of 9193.76 ms, indicating its superior performance in minimizing packet transit times and enhancing network responsiveness in B-WSN scenarios. These results underscore OctoRoute as the preferred routing protocol for reducing communication latency and ensuring timely data delivery in underwater sensor networks, optimizing overall network performance and reliability. Energy consumption is a crucial aspect in evaluating the performance of routing protocols in B-WSNs, as it directly impacts the network's operational lifespan and sustainability. Energy consumption refers to the power network nodes utilise to transmit and receive data packets. As depicted in Figure 4, OctoRoute consistently demonstrates lower energy consumption than MO-CBACORP and HOCOR across node densities. At a node density of 500, OctoRoute exhibits the lowest energy consumption of 62.451%, followed by HOCOR (81.888%) and MO-CBACORP (94.080 %). On average, OctoRoute maintains the lowest energy consumption of 55.309%, indicating its efficiency in optimizing power usage and prolonging the network's operational lifetime.

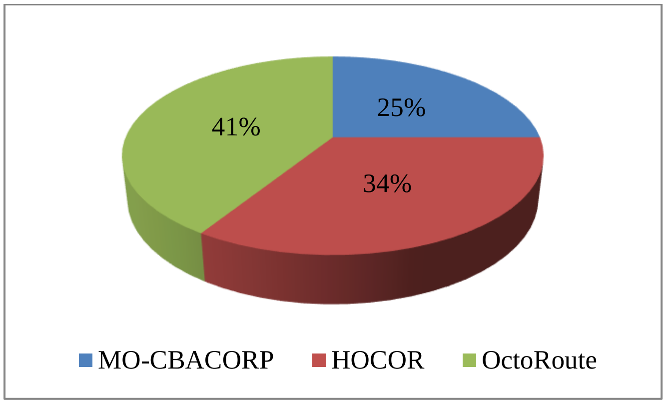

Fig. 2 Throughput

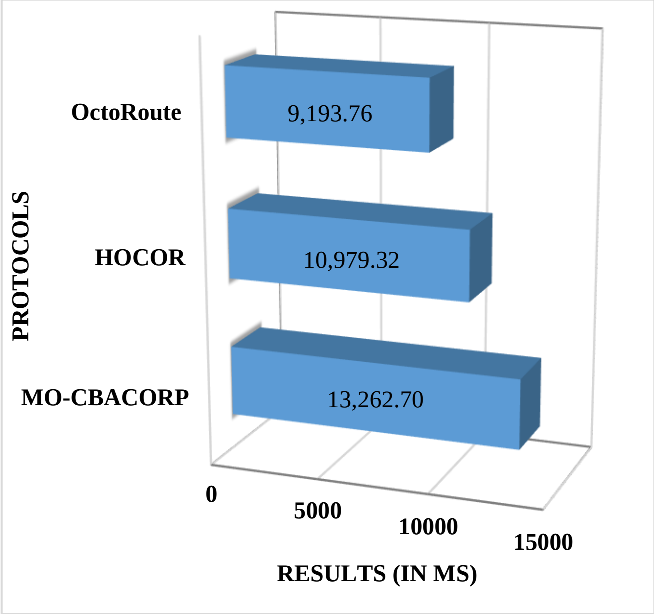

Fig. 3 Delay

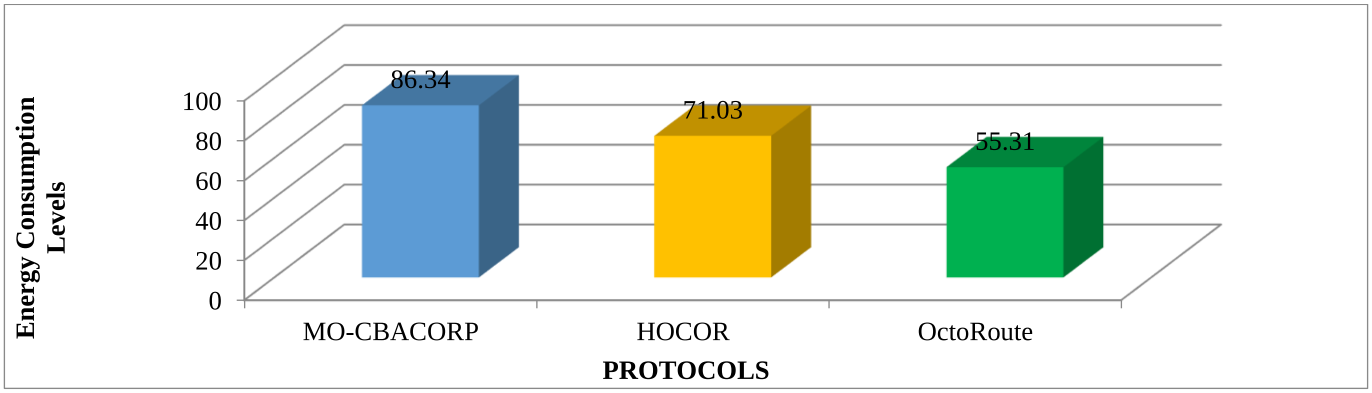

Fig. 4 Energy consumption

These findings highlight OctoRoute as the preferred routing protocol for minimizing energy consumption and maximizing the energy efficiency of underwater sensor networks. By selecting OctoRoute, network operators can effectively conserve energy resources and ensure prolonged network functionality, thereby enhancing the overall sustainability and reliability of B-WSN deployments.

6. Conclusion

Evaluating routing protocols in B-WSNs highlights OctoRoute as the superior choice for underwater communication. OctoRoute consistently outperforms MO-CBACORP and HOCOR across multiple performance metrics, including Packet Delivery Ratio (PDR), Packet Drop Ratio (PDPR), Throughput, Delay, and Energy Consumption. Its robust performance in delivering packets, maximizing throughput, minimizing delay, and optimizing energy usage underscores OctoRoute's effectiveness in real-world underwater scenarios. OctoRoute ensures reliable data transmission, efficient network operation, and prolonged lifespan of B-WSNs. These findings emphasize OctoRoute's significance in enhancing the performance, reliability, and sustainability of underwater sensor networks, thereby advancing the capabilities of underwater communication technology. OctoRoute is a promising solution for addressing the unique challenges of underwater communication and facilitating efficient data exchange in underwater environments.

References

[1] Monika Chaudhary et al., "Underwater Wireless Sensor Networks: Enabling Technologies for Node Deployment and Data Collection Challenges," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3500-3524, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Rucun Jia, "Design of A Simulation Platform for Intelligent Marine Search and Rescue Based on Wireless Sensors," *Microprocessors and Microsystems*, vol. 80, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Shuxian Su et al., "Application of Ship Electric Propulsion Based on Internet of Things System and Electronic System," *Microprocessors and Microsystems*, vol. 81, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Anwar Elhadad, and Seokheun Choi, "Powering the Internet of Things in Aquatic Environments: Solar Energy Harvesting Through a Buoyant Biosolar Cell Array," *Journal of Power Sources*, vol. 581, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Ganggang Liang et al., "Performance Enhancement of Hydrodynamic Piezoelectric Energy Harvester by Array Disturbance," *Energy Conversion and Management*, vol. 298, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Amrita Datta, and Mou Dasgupta, "Energy Efficient Topology Control in Underwater Wireless Sensor Networks," *Computers and Electrical Engineering*, vol. 105, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Aya Ayadi et al., "A Framework of Monitoring Water Pipeline Techniques Based on Sensors Technologies," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 2, pp. 47-57, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] K. Sumathi, and D. Vimal Kumar, "Ambient Intelligence-Based Fish Swarm Optimization Routing Protocol for Congestion Avoidance in Mobile Ad-Hoc Network," *International Journal of Computer Networks and Applications (IJCNA)*, vol. 9, no. 3, pp. 340-349, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] S. Kawsalya, and D. Vimal Kumar, "Invigorated Chameleon Swarm Optimization-Based Ad-Hoc On-Demand Distance Vector (ICSO-AODV) for Minimizing Energy Consumption in Healthcare Mobile Wireless Sensor Networks," *International Journal of Computer Networks and Applications (IJCNA)*, vol. 11, no. 2, pp. 191-212, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Saziya Tabbassum, and Rajesh Kumar Pathak, "Effective Data Transmission Through Energy-Efficient Clustering and Fuzzy-Based IDS Routing Approach in Wsns," *Virtual Reality and Intelligent Hardware*, vol. 6, no. 1, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Vishal B. Patil, and Surekha Kohle, "A High-Scalability and Low-Latency Cluster-Based Routing Protocol in Time-Sensitive Wsns Using Genetic Algorithm," *Measurement: Sensors*, vol. 31, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ines Lahmar et al., "Optimal Data Transmission for Decentralized IoT and WSN based on Type-2 Fuzzy Harris Hawks Optimization," *Internet of Things*, vol. 25, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Malathi Murugesan et al., "Enhancing Network Lifetime of WSNs through the Implementation of a Modified Ant Colony Optimization Algorithm," *Procedia Computer Science*, vol. 230, pp. 368-376, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Vasim Babu M. et al., "AE-LEACH: An Incremental Clustering Approach for Reducing the Energy Consumption in WSN," *Microprocessors and Microsystems*, vol. 93, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Xiaojun Mei et al., "Target Localization Using Information Fusion in Wsns-Based Marine Search and Rescue," *Alexandria Engineering Journal*, vol. 68, pp. 227-238, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] Zhongwei Shen et al., "A Power Control-Aided Q-Learning-Based Routing Protocol for Optical-Acoustic Hybrid Underwater Sensor Networks," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 4, pp. 2117-2129, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] Debing Wei et al., "Power-Efficient Data Collection Scheme for AUV-Assisted Magnetic Induction and Acoustic Hybrid Internet of Underwater Things," *IEEE Internet of Things Journal*, vol. 9, no. 14, pp. 11675-11684, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Shahrokh Vahabi et al., "CBDS2R: A Cluster-Based Depth Source Selection Routing for Underwater Wireless Sensor Network," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 9, pp. 468-476, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] Cangzhu Xu et al., "An Efficient Deployment Scheme with Network Performance Modeling for Underwater Wireless Sensor Networks," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 8345-8359, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[20] Chao Wang et al., "Multi-Agent Reinforcement Learning-Based Routing Protocol for Underwater Wireless Sensor Networks with Value of Information," *IEEE Sensors Journal*, vol. 24, no. 5, pp. 7042-7054, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Yanan Li et al., "An AUV-Assisted Data Collection Scheme for UWSNs Based on Reinforcement Learning Under the Influence of Ocean Current," *IEEE Sensors Journal*, vol. 24, no. 3, pp. 3960-3972, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Swati Gupta, and Niraj Pratap Singh, "Underwater Wireless Sensor Networks: A Review of Routing Protocols, Taxonomy, And Future Directions," *The Journal of Supercomputing*, vol. 80, pp. 5163-5196, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[23] Sangeeta Kumari, Pavan Kumar Mishra, and Veena Anand, "Coverage and Connectivity Aware Deployment Scheme for Autonomous Underwater Vehicles in Underwater Wireless Sensor Networks," *Wireless Personal Communications*, vol. 132, pp. 909-931, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Aifen Yao, and Jianmei Chen, "A Novel Method for Routing in Underwater Wireless Sensor Networks to Increase Efficiency Based on IoT," *Multiscale and Multidisciplinary Modeling, Experiments and Design*, vol. 7, pp. 411-424, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[25] Yangmei Zhang, Zhouzhou Liu, and Yang Bi, "Node Deployment Optimization of Underwater Wireless Sensor Networks Using Intelligent Optimization Algorithm and Robot Collaboration," *Scientific Reports*, vol. 13, no. 1, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Ashwini B. Gavali, Vinod M. Vaze, and Swapnaja A. Ubale, "HOCOR: Hybrid Optimization-Based Cooperative Opportunistic Routing for Underwater Wireless Sensor Networks," Wireless Personal Communications, vol. 135, pp. 1449-1472, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[27] Mingyue Zhang et al., "MO-CBACORP: A New Energy-Efficient Secure Routing Protocol for Underwater Monitoring Wireless Sensor Network," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 9, 2023. [CrossRef] [Google Scholar] [Publisher Link]