XVII Jornadas de Ingeniería Telemática
JITEL 2025
Universidad de Extremadura

Actas de las XVII Jornadas de Ingeniería Telemática (JITEL 2025), Cáceres (España), 12-14 de noviembre de 2025.

ISBN

A Reinforcement Learning-Based Telematic Routing Protocol for the Internet of Underwater Things

Mohammadhossein Homaei[*], Mehran Tarif[†], Agustin Di Bartolo[*], Óscar Mogollón Gutierrez[*], Mar Avila[*]
[*]Department of Computer Systems Engineering and Telematics, University of Extremadura,
Cáceres, 10003, Extremadura, Spain
Email: mhomaein@alumnos.unex.es, adibartolo@unex.es, oscarmg@unex.es, mmavila@unex.es
[†]Department of Computer Science, University of Verona, 37134 Verona, Italy
Email: mehran.tarifhokmabadi@univr.it

The Internet of Underwater Things (IoUT) faces major challenges such as low bandwidth, high latency, mobility, and limited energy resources. Traditional routing protocols like RPL, which were designed for land-based networks, do not perform well in these underwater conditions. This paper introduces RL-RPL-UA, a new routing protocol that uses reinforcement learning to improve performance in underwater environments. Each node includes a lightweight RL agent that selects the best parent node based on local information such as packet delivery ratio, buffer level, link quality, and remaining energy. RL-RPL-UA keeps full compatibility with standard RPL messages and adds a dynamic objective function to support real-time decision-making. Simulations using Aqua-Sim show that RL-RPL-UA increases packet delivery by up to 9.2%, reduces energy use per packet by 14.8%, and extends network lifetime by 80 seconds compared to traditional methods. These results suggest that RL-RPL-UA is a promising and energy-efficient routing solution for underwater networks.

***Keywords*—Internet of Underwater Things, Reinforcement Learning, RPL, Adaptive Routing, Energy Efficiency.**

I. Introduction

The IoUT is becoming a key technology for applications such as marine environmental monitoring, underwater exploration, and offshore infrastructure inspection. These systems use acoustic sensor networks that operate in harsh environments with high latency, unstable connections, and strict limitations on energy and bandwidth [1], [2]. Unlike terrestrial IoT, underwater networks rely on acoustic signals, which are slower and less reliable than radio waves. Nodes are often battery-powered and difficult to recharge, making energy efficiency critical. Furthermore, network topology can change due to node mobility, making routing and data delivery unpredictable. These challenges demand adaptive and lightweight communication protocols.

The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) is widely used in terrestrial IoT. It builds routing trees called Destination-Oriented Directed Acyclic Graphs (DODAGs) based on metrics like hop count or Expected Transmission Count (ETX). While RPL is efficient in static and low-power settings, its static behavior and limited reactivity make it unsuitable for dynamic and delay-prone underwater environments. In our previous work [3], we adapted RPL for underwater use by modifying its objective function (OF) and communication logic to better match acoustic conditions.

These limitations highlight the need for a routing solution that is both adaptive and energy-aware, while remaining compatible with existing RPL mechanisms. Our motivation is to design such a solution using reinforcement learning, enabling underwater nodes to react to dynamic conditions without high processing or communication costs.

Building on our previous work [3], this paper presents RL-RPL-UA, a RL-enhanced version of RPL for IoUT. Each node runs a lightweight RL agent that selects the next hop based on local observations such as energy level, link quality, buffer status, and delivery history. The protocol uses a composite, tunable OF to guide parent selection while maintaining full compatibility with RPL control messages (DIO/DAO). RL has proven effective for adaptive routing in dynamic networks, as it allows nodes to improve decisions over time based on experience. However, most RL-based protocols either lack RPL compatibility or demand high resources. RL-RPL-UA addresses this by offering a scalable, resource-efficient solution that adapts to underwater conditions without modifying the

RPL protocol structure.

The rest of the paper is organized as follows: Section II reviews relevant routing protocols for IoUT. Section III describes the RL-RPL-UA architecture and objective function. Section IV presents the simulation setup and performance evaluation. Finally, Section V summarizes conclusions and outlines future work.

II. Related Works

High propagation delay, limited energy resources, frequent disconnections, and dynamic topologies brought on by node mobility are just a few of the particular physical and operational limitations that make IoUT networks extremely difficult to use. Many routing protocols have been proposed to address these problems. These fall into five main families: (i) clustered and depth-based protocols; (ii) game-theoretic and opportunistic tactics; (iii) AI- and RL-driven methods; (iv) bio-inspired and meta-heuristic algorithms; and (v) extensions based on RPL.

A. Depth-Based and Clustered Routing

By grouping nodes into clusters or using their depth information, depth-based and clustered routing techniques seek to minimise latency and energy consumption. By utilising node depth and adaptive cluster formation, early protocols like C-GCo-DRAR [4] and U-(ACH)^2 [5] reduce latency and transmission overhead. To maximise cluster-head selection and increase network lifetime, FLCEER uses fuzzy logic [6]. Whereas IDA-OEP incorporates intelligent data analytics for energy-aware forwarding [1], BES uses bald-eagle-search optimisation for energy efficiency [7]. These schemes are useful in situations that are mobile or extremely dynamic, but they are often inflexible and necessitate accurate environmental calibration, despite their effectiveness in static conditions.

B. Opportunistic and Game-Theoretic Methods

Protocols that are opportunistic and game-theoretic address void zones, improve dependability, and use less energy. For instance, in 3-D acoustic networks, GTRP uses Nash equilibria to control relay selection [8]. While PCR [9] dynamically modifies gearbox power, hybrid solutions such as A-ANTD [10] and TARD [2] rely on autonomous underwater vehicles (AUVs) for delay-tolerant data collection. Although these designs improve performance in certain deployments, they typically rely on extensive pre-configuration or centralised control, which prevents large-scale autonomy.

C. AI and RL Approaches

AI is being used in recent work to accomplish self-adaptive routing. Li *et al.* employ multi-agent RL for optical IoUT links [11], Khan *et al.* use Q-learning for void mitigation [8], and Nandyala *et al.* develop topology-aware Q-routing [12], [13]. To stabilise paths under mobility, Tarif *et al.* combine fuzzy inference [14], [15]. Nearest to our focus, Tarif *et al.* (2025) present UWF-RPL, a fuzzy-logic extension of RPL that weighs ETX, depth, residual energy, and latency in a Mamdani controller, achieving a 17% PDR gain and 15% energy savings over baseline RPL [16]. However, unlike the lightweight Q-learning agent used in our RL-RPL-UA, its rule base is static and it is unable to change weights online.

D. Meta-Heuristic and Bio-Inspired Algorithms

Swarm intelligence is used by bio-inspired and meta-heuristic methods like FFRP (Firefly) [17], EORO (enhanced PSO) [18], and BES [7] to select energy-efficient paths. A fuzzy region-based algorithm that accommodates sink mobility is proposed by Pradeep *et al.* [19]. Their scalability in real-world deployments is frequently limited by the requirement for global optimisation and the lack of continuous learning, despite their encouraging simulation results.

E. RPL-Based Extensions

Originally created for sensor networks on land, the Routing Protocol for Low-Power and Lossy Networks (RPL) has been modified for use in underwater environments. Many studies have concentrated on extending RPL to work with underwater acoustic communication because it is the standard protocol for terrestrial IoT. In UW/MRPL [3], we adapted RPL for underwater environments by incorporating depth-aware routing metrics and mobility support. Despite being an improvement over baseline protocols such as OF0 and MRHOF, it lacked real-time adaptability and employed fixed objective function (OF) weights. UWF-RPL [16] incorporated a fuzzy logic-based OF into standard RPL control messages (DIO/DAO) to address compatibility and energy balancing. It did not, however, employ feedback mechanisms to enhance routing choices, and membership functions still needed to be manually adjusted. By incorporating a RL agent that automatically modifies OF weights in real-time, our suggested RL-RPL-UA improves RPL while preserving full protocol compatibility and removing the need for manual configuration.

Table I

Comparison of Recent Routing Protocols with RL-RPL-UA

Protocol	**RL**	**RPL**	**Adaptive OF**	**Mobility**	**Citation**
C-GCo-DRAR	–	–	Static OF	Limited	[4]
FLCEER	–	–	Static OF	Moderate	[6]
IDA-OEP	–	–	Static OF	Limited	[1]
GTRP	–	–	Static OF	Moderate	[8]
RL Protocol	✓	–	Static OF	Moderate	[13]
Q-Learning	✓	–	Dynamic OF	Moderate	[12]
Multi-agent RL	✓	–	Static OF	High	[11]
UA-RPL	–	✓	Static OF	Moderate	[20]
URPL	–	✓	Dynamic OF	Moderate	[14]
Fuzzy-CR	–	✓	Decision Making	Moderate	[15]
UWF-RPL	–	✓	Static Fuzzy OF	Moderate	[16]
UW/MRPL (prev. work)	–	✓	Static OF	High	[3]
RL-RPL-UA (this work)	✓	✓	Dynamic	High	–

We briefly review them above, and use Tables I and II to illustrate why RL-RPL-UA is necessary and novel.

III. Proposed Protocol: RL-RPL-UA

In this section, we introduce RL-RPL-UA, a novel routing protocol that enhances the conventional RPL protocol by embedding an RL agent into each node of the

Table II
KEY DIFFERENCES BETWEEN UWF-RPL, UWMRPL, AND RL-RPL-UA

Feature	UWF-RPL [16]	UWMRPL [3] (Previous work)	RL-RPL-UA (This Work)
Main Concept	Fuzzy-logic RPL for optimized routing	Mobility-aware RPL with static tunable OF	RL-based dynamic routing with local agents
Routing Adaptability	Semi-adaptive via static fuzzy logic rules	Semi-adaptive via predefined logic	Fully adaptive via real-time RL updates
OF	Static fuzzy logic (depth, energy, latency, ETX)	Static/custom (ETX, depth)	Dynamic composite (energy, LQI, queue, PDR)
RPL Compatibility	Extended DIO/DAO with fuzzy logic metrics	Standard-compliant	Extended DIO/DAO with RL metrics
Learning Agent	None (static fuzzy logic)	None	Q-learning or DQN per node
Reward Mechanism	None	None	$\alpha \cdot$ PDR $- \beta \cdot$ Delay $- \gamma \cdot$ Cost
Overhead	Moderate (fuzzy logic computations)	Low (no learning updates)	Low (optimized RL updates)
Mobility Handling	Reactive via fuzzy logic evaluation	Reactive DAG repair	Proactive via learned feedback
Queue Management	Included (congestion control)	Not included	Included (adaptive queue management)
Energy Efficiency	Good (static optimized)	Moderate (no dynamic optimization)	High (real-time optimization)
Key Contribution	Improved stability and efficiency via fuzzy logic	Mobility and energy-aware extension of RPL	Online adaptive parent selection via RL

underwater IoT network. Unlike traditional RPL implementations that rely on static OFs, our model leverages an adaptive learning mechanism to select optimal routes under the harsh and dynamic conditions of underwater communication.

A. Protocol Architecture

The architecture of RL-RPL-UA integrates an RL agent within the standard RPL stack. Each node consists of the following modules:

- **Sensing Unit:** Gathers state information including residual energy, buffer occupancy, and signal strength.
- **Communication Module:** Interfaces with an acoustic modem or underwater simulation module (Aqua-Sim, NS-2 with UAN).
- **RL Agent:** A local Q-learning or DQN model.
- **Extended RPL Stack:** Supports modified DIO/DAO messages carrying dynamic state and learned metrics.

B. RL Model

The routing process is modeled as a Markov Decision Process (MDP), where each node learns an optimal routing policy by interacting with its environment.

1) State Space: The state s_t at time t includes:

$$s_t = [E_t, \mathrm{LQI}_t, Q_t, \mathrm{PDR}_t, T_t] \tag{1}$$

where E_t is residual energy, LQI_t is link quality indicator, Q_t is current queue size, PDR_t is historical packet delivery ratio, and T_t is time since last successful transmission.

2) Action Space: The action a_t is the selection of a next-hop parent from among n neighbors:

$$a_t \in \{\mathrm{Parent}_1, \mathrm{Parent}_2, \ldots, \mathrm{Parent}_n\} \tag{2}$$

As shown in Equation (2), the action space consists of the set of all neighboring nodes that can serve as the next hop in the routing process.

3) Reward Function: The reward signal r_t is defined to balance reliability, delay, and energy consumption:

$$r_t = \alpha \cdot \mathrm{PDR}_t - \beta \cdot \mathrm{Delay}_t - \gamma \cdot \mathrm{EnergyCost}_t \tag{3}$$

As shown in Equation (3), this formulation enables the agent to optimize routing decisions by weighing the positive effect of packet delivery against the negative impact of delay and energy consumption. Here, α, β, and γ are tunable hyperparameters that control the trade-offs between these objectives. This reward is used to update the RL agent's policy.

Algorithm 1 RL-enhanced RPL Routing

```
Require: Initialization of Q-table or DQN weights, neighbor table, default Rank
Ensure: Energy-efficient and adaptive routing in underwater IoT
Initialize RL agent (Q-table or DQN), default Rank
$s \leftarrow$ OBSERVE_STATE
Broadcast DIO with $OF_{RL}(n_i)$ and node state
while Node is active do
    Receive DIOs from neighbors
    for all neighbor $n_i$ in NeighborTable do
        Extract state features: $s_{n_i} = [E, LQI, Q, PDR, T]$
        Compute $OF_{RL}(n_i)$ using Equation 5
        Estimate $Q(s, a = n_i)$ using RL model (Q-table or DQN)
    end for
    Select Parent:
    $a^* \leftarrow \arg\max_{n_i} Q(s, a = n_i)$        ▷ Best next-hop based on RL
    Update RPL Rank based on selected parent and $OF_{RL}(a^*)$
    Forward data packets to $a^*$
    Wait for Acknowledgement or Timeout
    Observe outcome:
    Measure $\mathrm{PDR}_t$, $\mathrm{Delay}_t$, $\mathrm{EnergyCost}_t$
    Compute reward $r_t$ using Equation 3
    $s_{t+1} \leftarrow$ OBSERVE_STATE
    RL Update:
    if Using Q-learning then
        Update Q-table using Equation 4
    else if Using DQN then
        Store $(s_t, a^*, r_t, s_{t+1})$ in ReplayBuffer
        Train DQN via minibatch sampling
    end if
    $s_t \leftarrow s_{t+1}$
    Periodically broadcast updated DIO with new Rank and $OF_{RL}$
end while
function OBSERVE_STATE
    Measure local energy $E$, link quality $LQI$, buffer queue $Q$, packet success
  rate $PDR$, time since last ACK $T$
    return $[E, LQI, Q, PDR, T]$
end function
```

4) Policy Learning: The RL agent seeks to learn a policy $\pi(a|s)$ that maximizes the expected cumulative reward:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \eta \Big[r_t + \gamma \cdot \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \Big] \tag{4}$$

Equation 4 is the standard Q-learning update rule, where η is the learning rate and γ is the discount factor for future rewards.

C. Adaptive OF

To replace the static OFs in RPL, we define a composite and dynamic OF:

$$OF_{RL}(n_i) = w_1 \cdot E(n_i) + w_2 \cdot R(n_i) + w_3 \cdot Q(n_i) + w_4 \cdot PDR(n_i) \tag{5}$$

Where:

- $E(n_i)$: Normalized remaining energy of neighbor n_i,
- $R(n_i)$: Link reliability (e.g., inverse of ETX),
- $Q(n_i)$: Queue length or buffer utilization,
- $PDR(n_i)$: Historical delivery ratio,
- w_1 to w_4: Adaptive weights tuned by the RL agent.

This OF is broadcast in DIO messages, allowing each node to evaluate its neighbors and update its rank dynamically.

D. Routing Decision Process

The routing process in RL-RPL-UA involves the following steps:

1) DIO Exchange: Each node broadcasts its current state and OF_{RL} value using an extended DIO message.
2) Neighbor Table Update: On receiving DIOs, nodes update their neighbor tables and estimate Q-values.
3) Parent Selection: The parent with the highest Q-value is selected as the preferred next-hop.
4) Data Forwarding: Data packets are forwarded along the learned optimal path.
5) Learning Update: After each transmission, the node observes outcomes and updates its Q-table using Equation 4.

E. Underwater-Specific Enhancements

The following improvements are tailored to the underwater environment:

- Delay Estimation: Nodes estimate propagation delay based on distance and water temperature to better model the reward function.
- Energy-Aware Slot Scheduling: TDMA (Time Division Multiple Access) is used as a MAC protocol to assign non-overlapping time slots to nodes, reducing collisions and idle listening in underwater acoustic networks.
- Clustered Learning: In large networks, cluster heads can aggregate policies and periodically disseminate updates.

F. Compatibility and Overhead

RL-RPL-UA remains compatible with legacy RPL nodes by embedding new fields in the optional sections of RPL messages. In terms of complexity:

- The Q-learning implementation requires minimal computational resources and is suitable for constrained devices.
- Communication overhead is slightly increased due to additional state sharing, but overall packet retransmissions are reduced.

G. Security Considerations

RL-RPL-UA can be extended to support trust-aware routing by integrating reputation scores into the reward function, allowing the network to avoid compromised nodes.

H. Resource and Energy Cost Estimation

To evaluate the feasibility of deploying RL-RPL-UA in real-world IoUT scenarios, we estimate the energy and processing cost based on standard underwater sensor node specifications. We consider nodes equipped with low-power microcontrollers (e.g., MSP430, ARM Cortex-M4) and acoustic modems such as the WHOI Micromodem or EvoLogics S2C.

1) Energy Cost per Transmission: Assuming a transmission power of 0.5 W and transmission time of 1.5 seconds per packet, the energy cost per transmission is calculated as:

$$E_{tx} = P_{tx} \times t = 0.5 \times 1.5 = 0.75 \text{ J} \tag{6}$$

As shown in Equation (6), each data transmission consumes approximately 0.75 joules.

2) Energy Cost per RL Update: The Q-table update process requires approximately 500–1000 CPU cycles. For a 16 MHz processor operating at 1.8 V and 3 mA, the energy cost is given by:

$$E_{cpu} = V \times I \times \frac{\text{cycles}}{f} = 1.8 \times 0.003 \times \frac{1000}{16 \times 10^6} \approx 0.34 \mu\text{J} \tag{7}$$

According to Equation (7), the energy consumption for a single RL update is approximately 0.34 microjoules.

3) Memory and Storage: The Q-table of size $n \times a$ with 8-bit values, for example with 10 neighbors and 5 actions, requires approximately 50 bytes. This is feasible for microcontrollers with at least 32 KB of SRAM.

4) Discussion: Compared to traditional RPL, RL-RPL-UA introduces minimal computational overhead due to the small Q-table and low RL update cost. However, it improves energy efficiency by reducing retransmissions and adapting paths proactively.

IV. SIMULATION RESULTS

A. SIMULATION PARAMETERS

To assess the performance of the proposed RL-RPL-UA protocol, we conducted simulations using Aqua-Sim, an extension of the NS-2 (Network Simulator 2) framework specifically designed for underwater acoustic network environments. The simulated network consists of both static and mobile sensor nodes deployed within a 3D underwater space using acoustic communication links. Each node independently executes the RL-RPL-UA algorithm and exchanges routing information via modified DIO/DAO messages.

We compare RL-RPL-UA against several baseline protocols, including standard RPL (OF0), Q-learning-only approaches, and cluster-based routing. Furthermore, we include direct comparisons with Co-DRAR [4], UA-RPL [20], and our prior works UWF-RPL [16], and UW/MRPL [3], to assess improvements in adaptability, energy efficiency, and delivery reliability. Evaluation metrics include Packet Delivery Ratio (PDR), End-to-End Delay, Energy Consumption, Routing Overhead, and Network Lifetime.

The main simulation parameters are listed in Table III.

Table III
SIMULATION PARAMETERS FOR RL-RPL-UA EVALUATION

Parameter	Value
Simulation Area	300 × 300 × 300 m^3
Number of Nodes	50
Initial Energy per Node	5 J
Transmission Power	0.5 W
Acoustic Bandwidth	10 kHz
Propagation Speed	1500 m/s
MAC Protocol	TDMA
Routing Protocols	RL-RPL-UA, RPL (OF0), Q-Routing
RL Algorithm	Q-learning (tabular)
Learning Rate (η)	0.1
Discount Factor (γ)	0.9
Reward Weights (α, β, γ)	(1.0, 0.6, 0.4)
Simulation Time	1000 s
Traffic Model	CBR, 1 packet/10 s
Packet Size	64 Bytes
Mobility Model	Random Waypoint (0.1–0.3 m/s)

B. Packet Delivery Ratio

PDR is calculated over K simulation trials as [21]:

$$\mathrm{PDR}_{\mathrm{mean}} = \frac{1}{K} \sum_{k=1}^{K} \left(\frac{R_k}{S_k} \right) \times 100 \tag{8}$$

$$\sigma_{\mathrm{PDR}} = \sqrt{ \frac{1}{K-1} \sum_{k=1}^{K} \left(\frac{R_k}{S_k} - \mathrm{PDR}_{\mathrm{mean}} \right)^2 } \tag{9}$$

In the static scenario, RL-RPL-UA achieves a mean Packet Delivery Ratio (PDR) of 94.3% with a standard deviation of 1.7, outperforming UWF-RPL (89.2%, σ=2.2), UWRPL (85.1%, σ=3.0), UA-RPL (83.5%, σ=3.0), and Co-DRAR (81.2%, σ=2.8). The results show that while UWF-RPL enhances PDR over traditional RPL variants by using adaptive metrics, RL-RPL-UA delivers a further 5.1% improvement over UWF-RPL and 9.2% over UWRPL, confirming the impact of RL in static deployments.

In the mobile scenario, RL-RPL-UA maintains the highest delivery performance with a PDR of 92.8% (σ=1.9), surpassing UWF-RPL (90.5%, σ=2.0), UWMRPL (88.2%, σ=2.3), UA-RPL (80.2%, σ=3.1), and Co-DRAR (78.4%, σ=3.2). Although UWF-RPL narrows the performance gap in mobile conditions through fuzzy logic and energy-aware decisions, RL-RPL-UA outperforms all baselines, confirming that its real-time learning approach significantly improves delivery reliability under dynamic underwater environments.

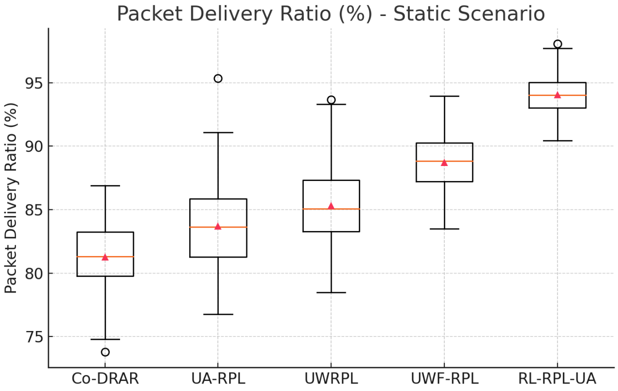

Fig. 1. PDR in the static scenario.

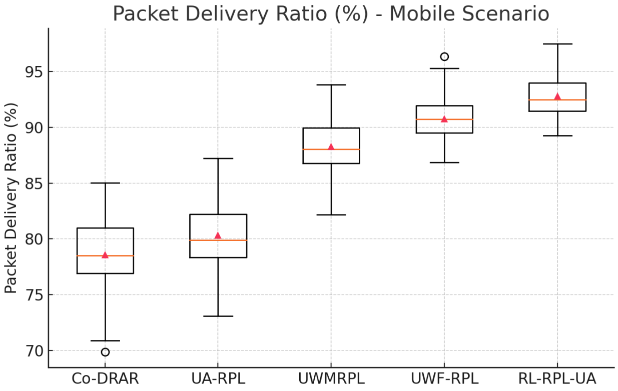

Fig. 2. PDR in the mobile scenario.

C. End-to-End Delay

The average packet delay in trial k is:

$$\mathrm{Delay}_k = \frac{1}{N_k} \sum_{j=1}^{N_k} \left(t_j^{\mathrm{recv}} - t_j^{\mathrm{send}} \right) \tag{10}$$

The overall mean and deviation:

$$\mathrm{Delay}_{\mathrm{mean}} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{Delay}_k \tag{11}$$

$$\sigma_{\mathrm{Delay}} = \sqrt{ \frac{1}{K-1} \sum_{k=1}^{K} (\mathrm{Delay}_k - \mathrm{Delay}_{\mathrm{mean}})^2 } \tag{12}$$

In the static scenario, RL-RPL-UA achieves an average end-to-end delay of 1.8 s (σ=0.2), outperforming UWF-RPL (2.0 s, σ=0.25), UWRPL (2.4 s, σ=0.3), UA-RPL (2.7 s, σ=0.4), and Co-DRAR (2.9 s, σ=0.4). The introduction of UWF-RPL demonstrates improvement over conventional RPL extensions, yet RL-RPL-UA further reduces delay by 10% compared to UWF-RPL and by 25% relative to UWRPL.

In the mobile scenario, RL-RPL-UA sustains low latency with an average delay of 1.9 s (σ=0.2), outperforming UWF-RPL (1.95 s, σ=0.25), UWMRPL (2.1 s, σ=0.3), UA-RPL (2.8 s, σ=0.4), and Co-DRAR (3.1 s, σ=0.4). These results highlight the RL agent's effectiveness in minimizing transmission delay under mobile and dynamically changing underwater network conditions.

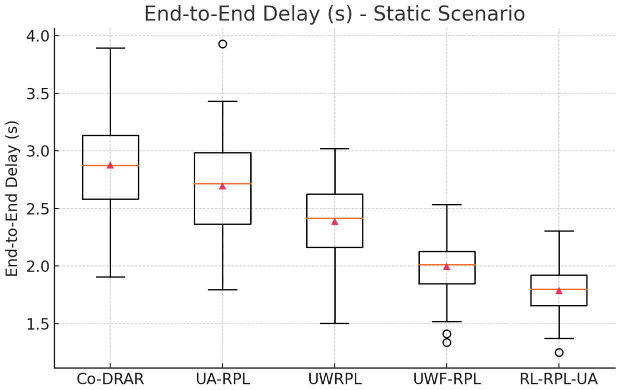

Fig. 3. End-to-End Delay in the static scenario.

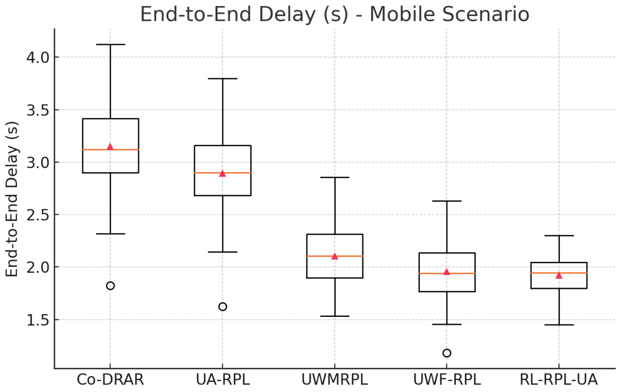

Fig. 4. End-to-End Delay in the mobile scenario.

D. Energy per Delivered Packet

Per trial, the energy cost per packet is:

$$\mathrm{E}_k = \frac{E_{\text{total},k}}{R_k} \tag{13}$$

Mean and deviation:

$$\mathrm{E}_{\text{mean}} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{E}_k \tag{14}$$

$$\sigma_{\mathrm{E}} = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}(\mathrm{E}_k - \mathrm{E}_{\text{mean}})^2} \tag{15}$$

In the static scenario, RL-RPL-UA achieves an average energy cost of 0.75 J per delivered packet (σ=0.05), significantly lower than UWF-RPL (0.78 J, σ=0.06), UWRPL (0.88 J, σ=0.07), UA-RPL (0.89 J, σ=0.07), and Co-DRAR (0.91 J, σ=0.08). While UWF-RPL improves energy efficiency by integrating adaptive cost metrics, RL-RPL-UA achieves an additional 3.8% energy saving over UWF-RPL and 14.8% over UWRPL, confirming its superior resource-awareness.

In the mobile scenario, RL-RPL-UA continues to deliver the most energy-efficient performance with an energy cost of 0.74 J (σ=0.05), followed by UWF-RPL (0.76 J, σ=0.055), UWMRPL (0.82 J, σ=0.06), UA-RPL (0.91 J, σ=0.08), and Co-DRAR (0.94 J, σ=0.09). These improvements reflect the effectiveness of the RL-based adaptive routing strategy in minimizing retransmissions and avoiding energy-intensive paths, even under dynamic network conditions.

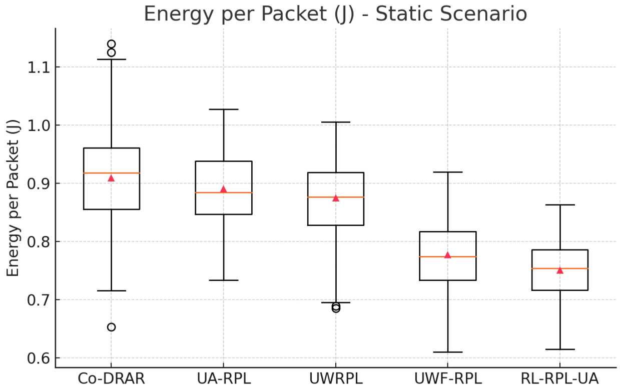

Fig. 5. Energy cost per delivered packet in the static scenario.

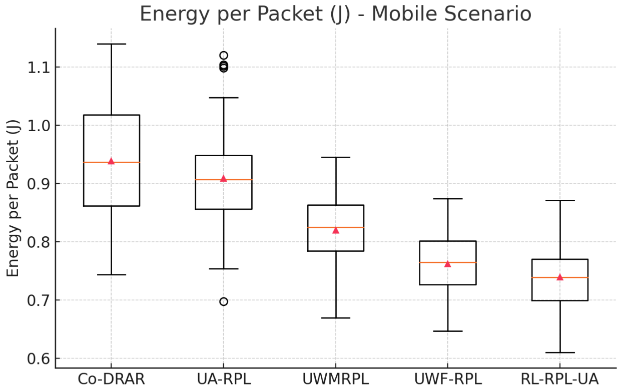

Fig. 6. Energy cost per delivered packet in the mobile scenario.

E. Routing Overhead Ratio

Overhead is computed as:

$$\mathrm{OH}_k = \frac{C_k}{R_k} \tag{16}$$

With:

$$\mathrm{OH}_{\mathrm{mean}} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{OH}_k \tag{17}$$

$$\sigma_{\mathrm{OH}} = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}(\mathrm{OH}_k - \mathrm{OH}_{\mathrm{mean}})^2} \tag{18}$$

In the static scenario, RL-RPL-UA introduces the lowest control overhead with a mean routing overhead ratio of 0.12 (σ=0.01), outperforming UWF-RPL (0.14, σ=0.015), UWRPL (0.22, σ=0.02), UA-RPL (0.24, σ=0.023), and Co-DRAR (0.25, σ=0.025). Although UWF-RPL reduces overhead compared to UWRPL and other classical protocols, RL-RPL-UA further reduces control traffic by 14.3% over UWF-RPL and 45% over UWRPL.

In the mobile scenario, RL-RPL-UA maintains minimal overhead at 0.11 (σ=0.01), followed by UWF-RPL (0.13, σ=0.012), UWMRPL (0.18, σ=0.015), UA-RPL (0.26, σ=0.028), and Co-DRAR (0.27, σ=0.03). Presented in consistent protocol order, these results confirm the effectiveness of RL-RPL-UA in suppressing control overhead even in dynamic, mobile environments.

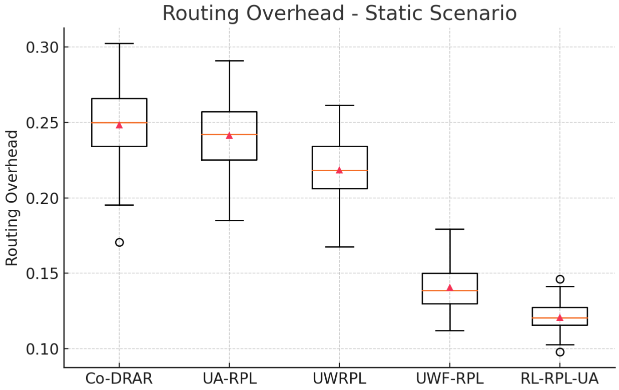

Fig. 7. Routing overhead ratio in the static scenario.

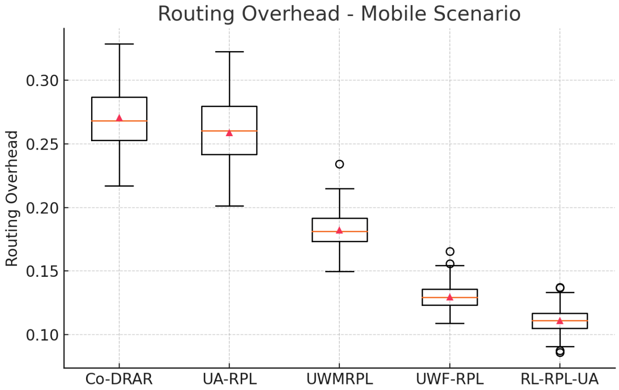

Fig. 8. Routing overhead ratio in the mobile scenario.

F. Network Lifetime

Lifetime is defined as the time until the first node in the network depletes its energy:

$$T^{\mathrm{death}}_{\mathrm{mean}} = \frac{1}{K}\sum_{k=1}^{K}T^{(k)}_{\mathrm{death}} \tag{19}$$

$$\sigma_T = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}\left(T^{(k)}_{\mathrm{death}} - T^{\mathrm{death}}_{\mathrm{mean}}\right)^2} \tag{20}$$

In the static scenario, RL-RPL-UA achieves the longest network lifetime of 720 seconds (σ=15), followed by UWF-RPL (690 s, σ=18), UWRPL (640 s, σ=22), UA-RPL (610 s, σ=24), and Co-DRAR (600 s, σ=25). The integration of fuzzy optimization in UWF-RPL enhances node longevity, but RL-RPL-UA further extends the lifetime by 30 seconds over UWF-RPL and 80 seconds over UWRPL, confirming the benefit of RL in energy-aware route planning.

In the mobile scenario, RL-RPL-UA sustains the longest network operation at 710 seconds (σ=16), ahead of UWF-RPL (700 s, σ=17), UWMRPL (680 s, σ=20), UA-RPL (590 s, σ=26), and Co-DRAR (580 s, σ=28). The improvement stems from RL-RPL-UA's ability to distribute energy consumption more evenly across nodes by dynamically selecting optimal, energy-efficient paths under varying underwater mobility conditions.

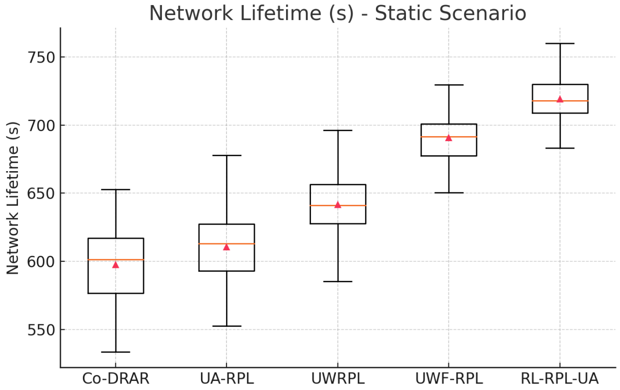

Fig. 9. Network lifetime (time until first node dies) in the static scenario.

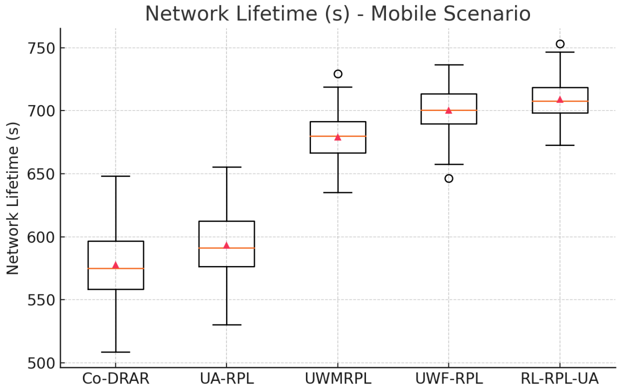

Fig. 10. Network lifetime (time until first node dies) in the mobile scenario.

V. Conclusion

This work presented RL-RPL-UA, a reinforcement learning-based extension of the RPL protocol designed for the challenges of the Internet of Underwater Things (IoUT). By incorporating Q-learning agents, the protocol adapts dynamically to changing network conditions and selects routing paths based on multiple performance criteria, including energy efficiency, link quality, queue length, and delivery reliability. The evaluation included a comprehensive comparison with recent baseline protocols under both static and mobile scenarios. The simulation results show that RL-RPL-UA offers consistent improvements in reliability, delay, energy consumption, control overhead, and network lifetime. These outcomes suggest that reinforcement learning can effectively enhance the adaptability and overall performance of routing protocols in underwater acoustic environments. Future research will focus on applying deep reinforcement learning to reduce training complexity and enable distributed decision-making among multiple agents in highly dynamic underwater networks.

Acknowledgment

Referencias

[1] Z. Wang, X. Gu, W. Xie, and D. Wu, "Qinghai Mutton Sales Mode Analysis and Optimization Strategy Research," *Procedia Computer Science*, vol. 242, pp. 1370–1377, 2024.

[2] K. Saleem, L. Wang, A. Almogren, E. Ntizikira, A. U. Rehman, S. Bharany, and S. Hussen, "Cognitive intelligence routing protocol for disaster management and underwater communication system in underwater acoustic network," *Scientific Reports*, vol. 15, no. 1, Mar. 2025.

[3] M. H. Homaei, A. J. Di Bártolo, R. Molano Gómez, P. G. Rodríguez, and A. Caro, "Enabling RPL on the Internet of Underwater Things," *Journal of Network and Systems Management*, vol. 33, no. 3, May 2025.

[4] Y. Guo, J. Jiang, Q. Yan, and G. Han, "An Opportunity Routing Protocol Based on Density Peaks Clustering in the Internet of Underwater Things," in *Proc. 2023 Int. Conf. on Intelligent Communication and Networking (ICN)*, pp. 175–179, Nov. 2023.

[5] M. Ismail, H. Qadir, F. A. Khan, S. Jan, Z. Wadud, and A. K. Bashir, "A novel routing protocol for underwater wireless sensor networks based on shifted energy efficiency and priority," *Computer Communications*, vol. 210, pp. 147–162, Oct. 2023.

[6] S. Natesan and R. Krishnan, "FLCEER: Fuzzy Logic Cluster-Based Energy Efficient Routing Protocol for Underwater Acoustic Sensor Network," *Int. J. of Information Technology and Web Engineering*, vol. 15, no. 3, pp. 76–101, Jul. 2020.

[7] N. Usman, O. Alfandi, S. Usman, A. M. Khattak, M. Awais, B. Hayat, and A. Sajid, "An Energy Efficient Routing Approach for IoT Enabled Underwater WSNs in Smart Cities," *Sensors*, vol. 20, no. 15, p. 4116, Jul. 2020.

[8] Z. A. Khan, O. A. Karim, S. Abbas, N. Javaid, Y. B. Zikria, and U. Tariq, "Q-learning based energy-efficient and void avoidance routing protocol for underwater acoustic sensor networks," *Computer Networks*, vol. 197, p. 108309, Oct. 2021.

[9] R. W. L. Coutinho, A. Boukerche, and A. A. F. Loureiro, "PCR: A Power Control-based Opportunistic Routing for Underwater Sensor Networks," in *Proc. 21st ACM Int. Conf. on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 173–180, Oct. 2018.

[10] Y. H. Robinson, S. Vimal, E. G. Julie, M. Khari, C. Expósito-Izquierdo, and J. Martínez, "Hybrid optimization routing management for autonomous underwater vehicle in the internet of underwater things," *Earth Science Informatics*, vol. 14, no. 1, pp. 441–456, Oct. 2020.

[11] X. Li, X. Hu, R. Zhang, and L. Yang, "Routing Protocol Design for Underwater Optical Wireless Sensor Networks: A Multiagent Reinforcement Learning Approach," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9805–9818, Oct. 2020.

[12] C. S. Nandyala, H.-W. Kim, and H.-S. Cho, "QTAR: A Q-learning-based topology-aware routing protocol for underwater wireless sensor networks," *Computer Networks*, vol. 222, p. 109562, Feb. 2023.

[13] İ. Eriş, Ö. M. Gül, and P. S. Bölük, "A novel reinforcement learning based routing algorithm for energy management in networks," *J. of Industrial and Management Optimization*, vol. 20, no. 12, pp. 3678–3696, 2024.

[14] M. Tarif and B. N. Moghadam, "Proposing a Dynamic Decision-Making Routing Method in Underwater Internet of Things," in *Proc. 2024 10th Int. Conf. on Artificial Intelligence and Robotics (QICAR)*, pp. 186–193, Feb. 2024.

[15] M. Tarif, M. Effatparvar, and B. N. Moghadam, "Enhancing Energy Efficiency of Underwater Sensor Network Routing Aiming to Achieve Reliability," in *Proc. 2024 Third Int. Conf. on Distributed Computing and High Performance Computing (DCHPC)*, pp. 1–7, May 2024.

[16] M. Tarif, M. H. Homaei, and A. Mosavi, "An Enhanced Fuzzy Routing Protocol for Energy Optimization in the Underwater Wireless Sensor Networks," *Computers, Materials & Continua*, vol. 83, no. 2, pp. 1791–1820, 2025.

[17] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "A Stateless Opportunistic Routing Protocol for Underwater Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 8237351, Jan. 2018.

[18] S. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "A Novel Cooperative Opportunistic Routing Scheme for Underwater Sensor Networks," *Sensors*, vol. 16, no. 3, p. 297, Feb. 2016.

[19] S. Pradeep, T. B. B. R. Bapu, R. Rajendran, and R. Anitha, "Energy Efficient Region based Source Distributed Routing Algorithm for Sink Mobility in Underwater Sensor Network," *Expert Systems with Applications*, vol. 233, p. 120941, Dec. 2023.

[20] Z. Liu, X. Jin, Y. Yang, K. Ma, and X. Guan, "Energy-Efficient Guiding-Network-Based Routing for Underwater Wireless Sensor Networks," *IEEE Internet of Things Journal*, vol. 9, no. 21, pp. 21702–21711, Nov. 2022.

[21] M. H. Homaei, S. S. Band, A. Pescapè, and A. Mosavi, "DDSLA-RPL: Dynamic Decision System Based on Learning Automata in the RPL Protocol for Achieving QoS," *IEEE Access*, vol. 9, pp. 63131–63148, 2021.