IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 29, 2021, accepted November 1, 2021, date of publication November 16, 2021,
date of current version November 24, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3128516

Reinforcement Learning-Based Routing Protocol for Underwater Wireless Sensor Networks: A Comparative Survey

REHENUMA TASNIM RODOSHI[1], (Graduate Student Member, IEEE), YUJAE SONG[2], (Member, IEEE), AND WOOYEOL CHOI[1], (Member, IEEE)
[1]Department of Computer Engineering, Chosun University, Gwangju 61452, Republic of Korea
[2]Maritime ICT Research and Development Center, Korea Institute of Ocean Science and Technology, Busan 49111, Republic of Korea

Corresponding author: Wooyeol Choi (wyc@chosun.ac.kr)

The work of Rehenuma Tasnim Rodoshi and Wooyeol Choi was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF-2021R1I1A3050535. The work of Yujae Song was supported in part by the project titled 'Development of Polar Region Communication Technology and Equipment for Internet of Extreme Things (IoET),' funded by the Ministry of Science and ICT (MSIT).

ABSTRACT Underwater wireless sensor networks (UWSNs) have emerged as a promising networking technology owing to their various underwater applications. Many applications require sensed data to be routed to a centralized location. However, the routing of sensor networks in underwater environments presents several challenges in terms of underwater infrastructure, including high energy consumption, narrow bandwidths, and longer propagation delays than other sensor networks. Efficient routing protocols play a vital role in this regard. Recently, reinforcement learning (RL)-based routing algorithms have been investigated by different researchers seeking to exploit the learning procedure via trial-and-error methods of RL. RL algorithms are capable of operating in underwater environments without prior knowledge of the infrastructure. This paper discusses all routing protocols proposed for RL-based UWSNs. The advantages, disadvantages, and suitable application areas are also mentioned. The protocols are compared in terms of the key ideas, RL designs, optimization criteria, and performance-evaluation techniques. Moreover, research challenges and outstanding research issues are also highlighted, to indicate future research directions.

INDEX TERMS Underwater wireless sensor network, routing protocol, reinforcement learning.

I. INTRODUCTION

Underwater wireless sensor networks (UWSNs) represent an emerging field in wireless communication, owing to their significant advantages in various underwater applications. A typical UWSN consists of several self-configurable sensor nodes anchored to the ocean floor; these are interconnected by automatically adaptive wireless links featuring one or more underwater gateways [1]. These sensor nodes are used to perform various tasks, including pollution monitoring, offshore oil-drill monitoring, disaster prevention, and geological event monitoring [2]. Moreover, different types of data (e.g., temperature, pressure, and chemical compositions for water-based-disaster warning, underwater military communications, and surveillance systems) can also be collected using UWSNs. However, these networks are considered a more challenging wireless communication medium than wired or wireless terrestrial ones. The marine environment exhibits several distinctive features that differ from those of the atmospheric environment in which traditional communication is performed. A simple USWN architecture featuring sensor nodes, sink nodes, and a base station is shown in Figure 1. The sensor nodes transmit data to the sink nodes using other sensor nodes as relays, according to different parameters. Then, the sink nodes send these data to the base station on the ocean surface. Sensor nodes are deployed at different depths (with respect to the surface) and at different distances from each other underwater. Some nodes are anchored to the ocean floor, whilst others float in the water at various depths. The node density may vary according to the necessity and application of nodes in different locations.

Four types of underwater communications for UWSNs are commonly employed in different research works: radio frequency, acoustic, optical, and magnetic induction.

The associate editor coordinating the review of this manuscript and approving it for publication was Hongwei Du.

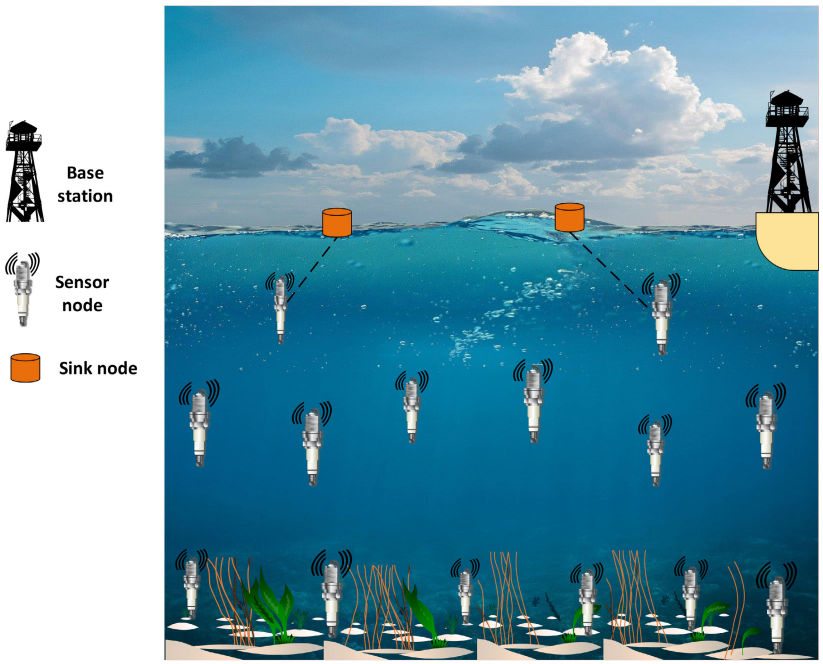

FIGURE 1. Simple UWSN architecture with sensor nodes, sink nodes, and base station.

The properties, advantages, and disadvantages of these four communication modes are compared in Table 1.

- **Radio frequency communication:** Terrestrial wireless sensor networks employ different electromagnetic waves for communication. Radio frequency electromagnetic waves represent an appropriate option for UWSNs when high data rates are required over a short communication range [3]. The data rates vary between freshwater and seawater when using electromagnetic communication. The propagation of radio waves differs from atmospheric propagation, owing to the high permittivity and electrical conductivity of water [4]. Radio frequencies are unfeasible for long-range underwater communication because they suffer from high attenuation, which severely limits their communication range [5]. However, radio-frequency communications have a range of applications in short-range navigation, sensing, and communications.
- **Acoustic communication:** In underwater acoustic communication, transmission and reception are realized using sound waves. This is by far the most commonly used method for underwater sensing and data transmission [6]. Acoustic communication is more attractive than other communication mediums because it can achieve large communication ranges of up to 20 km [7]. The underwater acoustic channel has three main characteristics: an attenuation that increases with signal frequency, a low sound speed, and multipath propagation [8]. Acoustic propagation performs better at low frequencies (10–15 kHz), and the total available bandwidth (5 kHz) is limited. High-speed acoustic communication in UWSNs is challenging, owing to the limited bandwidth, high transmission losses, multipath fading, and large Doppler shifts [9]. Moreover, acoustic communication in shallow water channels is more difficult than in deep-water communication.
- **Optical communication:** Optical USWN communication offers the highest data rates (up to Gbps), lowest delays, and smallest implementation costs when compared to other underwater communication approaches [10]. The high propagation speed (i.e., the speed of light) can improve its applicability in real-time underwater tasks. However, it suffers from absorption and scattering problems, attributable to the underwater characteristics and environment [11]. In optical communication, signal attenuation and scattering degrade the data transmission quality over long distance [12]. Owing to the narrow divergence properties of light, several researchers have employed LEDs or blue/green lasers as light sources; however, these require precise alignment [13]. The communication range of underwater optical communications (10–150 m) must be extended to appropriately implement real-time monitoring applications.
- **Magnetic induction communication:** UWSNs employing magnetic-induction-based communication are a relatively new communication paradigm compared to others. In this technology, a time-varying magnetic field is employed for data transmission [14]. This offers low latency, predictable channel behavior, long communication ranges (with large bandwidths), and silent and stealth applications in underwater environments.

TABLE 1. Comparison among different communication techniques for UWSNs.

Parameters	Communication range	Data rate	Propagation speed	Latency	Advantages	Disadvantages
Radio frequency	Up to 10 m	$\sim Mbps$	2.3×10^8 m/s	Moderate	- High data rates - Robust against acoustic noise - Unaffected by multipath - No known effects on underwater creatures	- Heavy attenuation - Susceptible to electromagnetic interference - Short communication range
Acoustic	Up to 20 km	$\sim Kbps$	1500 m/s	High	- Mostly adopted technique - Long communication range - Energy efficient - Precision navigation	- Low bandwidth. - High transmission losses - Time-varying multipath propagation - High latency and Doppler spread - Poor in shallow water - Ambient noise and site-specific noise
Optical	10-150 m	$\sim Gbps$	2.3×10^8 m/s	Low	- Low energy consumption - High data rates over short transmission ranges - High bandwidth - Simpler computational complexities for short-range links - Low latency	- High absorption and scattering by optical beams - Significant engineering and maintenance issues
Magnetic induction	10-100 m	$\sim Mbps$	3.3×10^7 m/s	Low	- High data rates - Suitable communication distance - Robust against multipath and Doppler effects - Smaller propagation delay - Low cost	- Signal corruption by thermal noise - Unpredictable polarization loss of coil antenna

The cost of the coils used in magnetic induction is relatively low, which makes it a strong candidate for large-scale deployment in UWSNs. Moreover, this mode is not affected by multipath propagation or fading and is robust against acoustic noise [15]. However, the performance of magnetic induction systems in UWSNs is still being researched, especially with regard to the characterization of broadband and complex underwater magnetic induction channels in shallow and lossy water [16]. Practical applications in both shallow and deep water show fully connected multi-coil networks can be implemented using bandwidths of the order of tens of kHz for small and large coverage areas [17].

The routing protocol in all types of wireless sensor network (WSN) plays a major role when designing schemes to transmit data from the source to the destination nodes. However, routing in UWSNs is of particular importance. The major challenges include limited bandwidth capacity, multipath fading, propagation delay, high bit-error rates, and temporary loss of connectivity. Designing efficient routing protocols in UWSNs is crucial for the quick and secure transmission of collected data to the sink node on the ocean surface. Numerous UWSN routing algorithms have been reported in the literature. These protocols are proposed to improve the efficiency with respect to end-to-end delay, node mobility, network throughput, and energy consumption.

Reinforcement learning (RL) [18] is a subfield of machine learning (ML) that utilizes an agent to take decisions in an unknown environment. The agent in RL algorithms follows a policy based upon immediate rewards for actions taken. Along with other ML techniques, RL has been widely used to design routing protocols for different WSNs [19], [20]. RL algorithms can be employed to improve the routing performance in UWSNs, owing to the constrained environment and the limitations of the UWSN environment. Different parameters of UWSN routing (e.g., energy efficiency, latency, network lifetime, link quality, and packet delivery ratio) can be optimized by implementing the RL algorithm. Because RL algorithms learn through experience, they have the potential to improve the routing process under various objectives.

Considering the advantages of RL, many researchers have proposed RL-based routing protocols for UWSNs. However, more research is required to successfully integrate the RL concept into the UWSN routing mechanism. In this regard, a comprehensive review paper presenting the existing RL-based routing protocols can help researchers seeking to design an RL-based UWSN routing protocol. In addition to filling the research gap in the literature, a survey on RL-based routing in UWSNs is required to encourage researchers to increase their focus on intelligent UWSN routing protocols.

Numerous survey works in the literature have compared different proposed routing protocols for UWSNs [21]–[24]. They divided and categorized the existing routing protocols according to different objectives. However, none of the existing surveys focused solely on RL-based routing protocols for UWSNs, despite numerous studies on this topic. To fill this gap in the literature and provide a direction for future research, it is necessary to aggregate these disparate works. Thus, a comparative study is necessary for RL-based UWSN routing. The main contributions of this study are as follows:

TABLE 2. Existing surveys on UWSN routing and RL-based routing.

Topic(s) of survey	Year of publication	Ref.	Routing in UWSN	RL-based routing protocols
Routing in UWSN	2011	[24]	Yes	No
Delay-tolerant UWSN routing	2014	[25]	Yes	No
Routing in UWSN	2015	[21]	Yes	No
Routing in UASN	2016	[26]	Yes	No
MAC and routing protocols in UWSN	2016	[27]	Yes	Not all discussed
Node mobility-based routing in UWSN	2017	[28]	Yes	No
Routing in UWSN	2017	[23]	Yes	No
Routing in UWSN	2018	[29]	Yes	No
Routing in UASN	2019	[22]	Yes	No
Requirements and recent advances in UWSN	2020	[30]	Yes	No
Energy efficient routing in UWSN	2021	[31]	Yes	Not all discussed
Localization in UWSN	2019	[1]	Not all discussed	No
RL-based routing in communication networks	2019	[32]	Not all discussed	Yes
RL-based routing in MANET	2011	[33]	No	Yes
RL-based routing in VANET	2021	[34]	No	Yes
RL-based routing in CRAHN	2019	[35]	No	Yes

- A brief overview of RL is presented, to provide a fundamental understanding of the technique.
- The existing RL-based UWSN routing protocols are investigated and summarized along with their advantages, disadvantages, and suitable applications in USWN environments.
- A comparative study of all the reviewed protocols is presented. In this regard, the key ideas of all protocols are compared in a tabular format. Then, a comparison of the applied RL techniques is provided.
- The optimization parameters adopted in all protocols are compared. The performance evaluation techniques are also compared in terms of the simulation environment, techniques, and performance comparisons of all the reviewed schemes.
- The key research challenges are highlighted, along with promising research directions toward making RL-based UWSN routing protocols more efficient.

The remainder of this paper is organized as follows. Section II describes existing related surveys in the literature, to highlight the necessity of the present survey. An overview of the RL technique is provided in Section III. III. All existing RL-based routing protocols for UWSNs are discussed in Section IV. In Section V, comparisons of the reviewed protocols are discussed according to their key ideas, optimization criteria, RL features, and performance measurement techniques. Challenges and open research issues are discussed in Section VI. Finally, Section VII concludes the paper.

II. EXISTING SURVEYS

This section describes the existing surveys regarding routing protocols for UWSNs and RL-based routing protocols for other WSN environments. The limitations of the existing works and the contributions of our study are also discussed. The existing surveys relating to UWSN and RL-based routing protocols are compared in Table 2.

Several surveys have been performed regarding UWSN routing protocols, focusing on different issues (e.g., energy efficient routing, node mobility, delay tolerant routing, and network-lifetime-aware routing). In [24], routing issues in UWSNs were discussed in terms of delivery ratio, end-to-end delay, energy efficiency, delay tolerant applications, mobility, and reliable routing. All routing protocols proposed thus far were also described. Cho *et al.* studied routing protocols considering the delay/disruption tolerance characteristics of UWSNs [25]. In this regard, they categorized the routing protocols into scheduled, opportunistic, and predicted contact schemes.

Han *et al.* classified UWSN routing protocols into sender-based and receiver-based protocols [21]. The protocols were then compared in terms of energy efficiency, latency, load balancing, dynamic robustness, communication overhead, and time complexity. In [26], UWSN routing protocols for acoustic communication were studied. The protocols were categorized using the cross-layer and non-cross-layer design methods. An intelligent algorithm for UWSN routing was also discussed. However, none of the RL-based routing protocols for UWSNs were mentioned. Unlike that, the authors in [27] discussed several RL-based routing protocols whilst studying the routing and medium access control (MAC) protocols for UWSNs. Their main aim was to quantitatively compare the existing MAC and routing protocols in terms of energy efficiency and reliability.

In [28], the routing protocols were studied by considering the node mobility in UWSNs. In this regard, the protocols are classified into vector-based, cluster-based, autonomous underwater vehicle (AUV)-based, depth-based, and path-based routing protocols. Both qualitative and quantitative comparisons between existing protocols were performed. Khalid *et al.* discussed the routing issues in UWSNs [23] whilst classifying the protocols into localization-based and localization-free routing protocols.

All protocols were described and compared in terms of the employed technique, as well as other important performance metrics. The authors in [29] conducted a simulation-based survey on UWSN routing protocols. Four routing protocols, namely hop-by-hop dynamic address-based routing, depth-based routing, energy-aware opportunistic routing, and energy-efficient depth-based routing, were implemented. The performances were compared in terms of the numbers of sent packets, alive nodes, and dead nodes.

Considering the acoustic communication in UWSNs, the routing protocols are reviewed in [22]. All protocols are categorized into localization-based and localization-free routing protocols. Moreover, each of the protocols was summarized whilst mentioning their strengths and weaknesses. A survey on different aspects of UWSNs was provided in [30]. The requirements of UWSNs (e.g., longevity, accessibility, complexity, security, and environmental sustainability) are highlighted. Moreover, the routing protocols are discussed alongside other issues in the UWSN. The authors in [31] discussed energy-efficient routing protocols for UWSNs. The protocols were categorized into depth-based, cluster-based, cooperation-reliability-based, RL-based, and bio-inspired routing protocols. However, only three URL-based UWSN routing protocols were mentioned. Unlike other surveys, the authors in [1] discussed UWSNs, focusing on both acoustic and magneto-inductive communication. They discussed the characteristics and application properties of each communication channel when designing UWSN routing protocols.

Considering the advantages of RL algorithms in routing protocol design, RL-based routing has been extensively studied in the literature. In [32], RL-based routing protocols for different types of communication networks were reviewed. The network areas considered were wired networks, wireless networks, wireless mesh networks, cooperative communication wireless networks, optical networks, ad-hoc networks, WSNs, vehicular ad hoc networks (VANETs), delay-tolerant networks (DTNs), social DTNs, flying ad hoc networks, cognitive radio networks, named-data networking, peer-to-peer networks, and software-defined networks. Several related surveys were also conducted for mobile ad hoc networks (MANETs) [33], cognitive radio ad-hoc networks (CRAHNs) [35], and VANETs [34]. A comprehensive survey on RL-based routing protocols in MANETs is provided along with future research directions in [33]. The authors in [34] extensively surveyed RL-based routing protocols for VANETs, by discussing their working process, advantages, limitations, and suitable application areas. Furthermore, the protocols were compared according to their main features, characteristics, evaluation methods, optimization criteria, and RL implementation. In [35], the RL-based efficient spectrum-aware routing for CRAHN was extensively discussed. Moreover, a multi-objective spectrum-aware routing protocol using RL was proposed to increase the probability of successful transmission with a minimum number of hops.

However, from the above-mentioned surveys in the literature, it is clear that no survey solely discusses RL-based routing protocols for UWSNs. The suitability of RL algorithms for solving optimization problems related to UWSN routing necessitates a survey that discusses all the studies in the literature. This will provide future researchers with an idea of the work already conducted, as well as potential research challenges and directions.

III. REINFORCEMENT LEARNING OVERVIEW

This section provides a brief overview of RL, by discussing the designs and classification of RL algorithms. RL is a sub-branch of ML. In RL, an agent learns by interacting with the environment and selects action based upon that learning. The learning process is similar to learning in the real world. The concept of RL seems straightforward, because it reflects the real world; however, implementing an RL algorithm can be a complex and challenging task. Such algorithms manage learning through interactions and feedback mechanisms; that is, learning to solve a problem using a trial-and-error approach.

A. MODELING OF RL ALGORITHM

The agent observes the state of the environment during each decision step, and it selects actions randomly or by following a policy. Next, it receives an immediate reward based upon the selected action and goes on to the next state. The reward function is designed to provide feedback to the learning algorithm, reflecting the primary objective of the task. The principle idea of RL is illustrated in Figure 2. There, the agent observes state s_t from the environment. In that particular state, the agent chooses action a_t by exploration or exploitation. According to the taken action, the agent receives a reward r_t and goes to the next state.

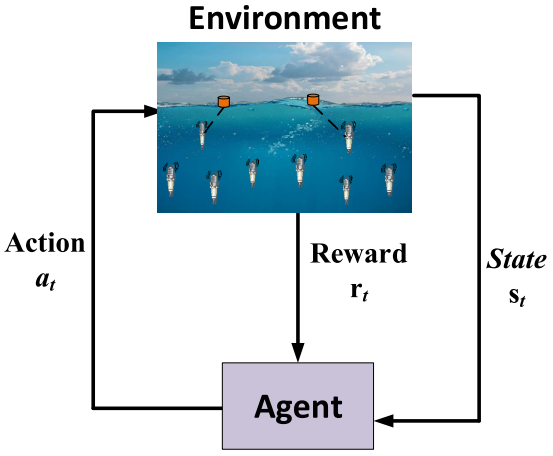

FIGURE 2. Basic working procedure of the RL algorithm.

To solve a problem with the help of RL, the problem should be designed as a Markov decision process (MDP) [36]. Therefore, MDP can be regarded as the theoretical basis of RL. The mathematical framework of MDP consists of a tuple of $< S, A, P, R >$, where S is a finite set of environment states, A is a set of actions available for the agent, P is the

transition probability from the current state to the next state via a particular action, and R is the reward received after transitioning to the next state with the taken action. The transition probability can be written as

$$P_a(s, s') = P_r(s_{t+1} = s' | s_t = s, a_t = a), \tag{1}$$

where P_a is the probability of transitioning from state s at time t to state s' at time $t + 1$ by taking action a. After the transition from s to s', the agent receives an immediate reward, which can be denoted by $R_a(s, s')$. The reward represents an evaluation of the quality of an action in a particular state.

The goal of an RL agent is to identify a policy π that maximizes the cumulative rewards; typically, this is the expected discounted sum of rewards. The policy is a function that maps a given state to the probability of selecting each possible action from that state. Thus, following a policy π, the probability of taking action a in state s at time t can be denoted by $\pi(a|s)$. The function that estimates how desirable it is for an agent to be in a given state, or how desirable it is to select a particular action in a given state, is called a value function. The value function can be a function of state or of state–action pairs.

The state value function $V_\pi(s)$ determines the value of a state for an agent following policy π. The value of a state s is the expected sum of discounted rewards starting from state s at time t following policy π. The value function can be written as

$$V_\pi(s) = E[R] = E[\sum_{t=0}^{\infty} \gamma^t r_t | s_t = s], \tag{2}$$

where R is a random variable defined as the sum of the future discounted rewards. It can be written as

$$R = \sum_{t=0}^{\infty} \gamma^t r_t, \tag{3}$$

where r_t is the reward at time t, and γ is a discount factor designed such that $0 \leq \gamma \leq 1$. The value of γ determines the importance of future rewards in the current state. Future rewards are discounted to place more emphasis on the immediate reward. The policy that optimizes the expected cumulative reward is referred to as the optimal policy and is denoted as π^*. An RL algorithm converges when it identifies the optimal policy from all available policies for a given state [18].

RL algorithms can be initially classified into model-based RL [37] and model-free RL [38]. Model-based RL algorithms construct an internal model describing the transitions and immediate outcomes according to experience. Then, the optimal policy for selecting an action is chosen using the learned model. However, model-free RL algorithms do not incorporate any learned models; learning is performed by either approximating value functions or following a policy through experiences. Therefore, RL algorithms can be designed using the policy or value iteration functions [34]. Examples of policy iteration-based RL include Monte Carlo [39] and temporal differencing methods [40].

In value-based RL algorithms, the agent attempts to maximize the value function. As mentioned earlier, the value functions in RL algorithms are of two types: state-value and action-value. The value function given in Equation 2 is the state-value function, which estimates the expected cumulative reward when starting in state s and following policy π thereafter. The action-value function denoted as $Q_\pi(s, a)$ determines the expected reward when action a is taken in a given state s following policy π. It can be defined as

$$Q_\pi(s, a) = E[\sum_{t=0}^{\infty} \gamma^t r_t | s_t = s, a_t = a]. \tag{4}$$

The action-value function Q_π is conventionally called the Q-function, and the output from this function for any given state–action pair is called the Q-value.

However, RL algorithms suffer from an exploration–exploitation trade-off when taking an action. On the one hand, the algorithms should not stick to the actions with high rewards, because they might thereby become trapped in a local optimum; on the other hand, repeatedly taking different actions from a single state is also inefficient. Different methods have been proposed to solve this problem, including random action [41], greedy strategy [42], epsilon-greedy policy [43], upper confidence bound [44], explore-first [45], and Softmax action [46].

B. ADVANTAGES OF USING RL IN UWSN ROUTING PROTOCOLS

In recent years, RL has been applied to design protocols in different wireless sensor networks; UWSN is one of them. The advantages of using RL for designing UWSN routing protocols are as follows:

- Routing optimization: RL algorithms can solve optimization problems in different distributed systems. Routing problem optimization can be regarded as a decision-making task. Therefore, RL can represent a practical approach for solving routing problems. In particular, solving routing problems with RL can be effective because of the reduced overheads for control packets, memory, and computation.
- Environment observability: In a UWSN, full information and knowledge of the network are unavailable. The RL algorithm can be effectively applied in such scenarios, because RL learns from the environment.
- Adapting to dynamic topology: RL-based routing learns the network topology whilst relaying packets. Hence, it can adapt to the dynamic network topology during the routing process. Moreover, RL algorithms learn iteratively, which helps reduce communication and computation overheads.

IV. RL-BASED ROUTING PROTOCOLS FOR UWSN

In this section, RL-based UWSN routing algorithms are discussed with respect to their working procedures. The advantages and disadvantages of each protocol are discussed, and

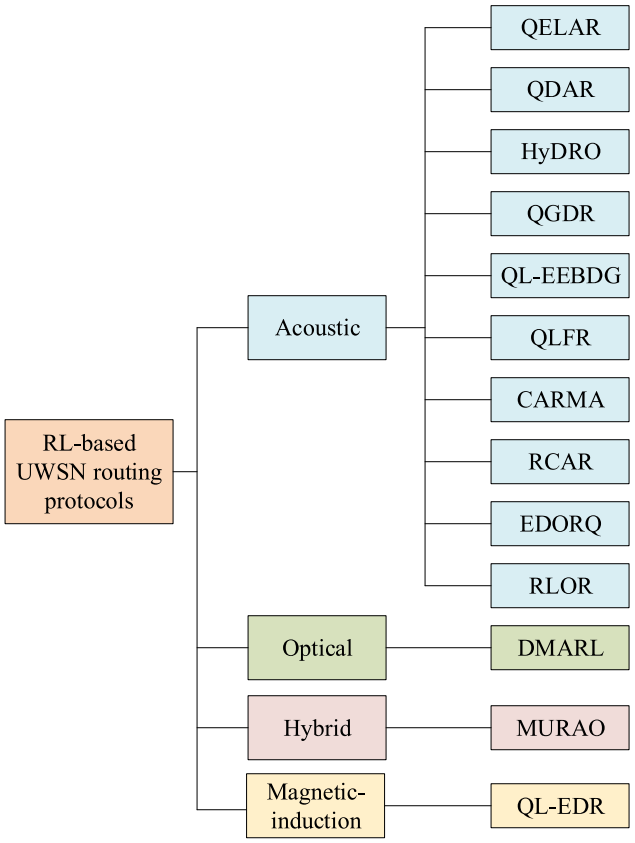

FIGURE 3. Categorization of RL-based UWSN routing protocols based on their communication medium.

suitable application areas based on the proposed scheme are highlighted. These routing protocols are categorized based on their UWSN communication medium: acoustic, optical, hybrid (acoustic–optical), and magnetic-induction. Figure 3 shows the taxonomy of the investigated routing protocols. The majority of protocols considered acoustic communication in UWSNs.

A. Q-LEARNING-BASED ENERGY-EFFICIENT LIFETIME-AWARE ADAPTIVE ROUTING (QELAR)

Hu *et al.* proposed QELAR [47], a distributed UWSN routing protocol that initially applies Q-learning to balance the workload between sensor nodes and thereby increase network lifetime and reduce network overhead. QELAR is an older UWSN routing protocol compared to the other protocols reviewed in this survey. In QELAR, when a node receives or overhears a packet, it extracts information from the packet header, including the residual energy, average group energy, previous-hop node, and V-value. The V-value of the node is calculated using the Q-learning algorithm. Once the Q-values of state–action pairs in a state s_n have been calculated for all available actions, another value function (the V-value) is calculated. The V-value of a state s_n is denoted by $V(s_n)$ and contains the maximum Q-value received by an action out of all actions in that state. Therefore, the V value can

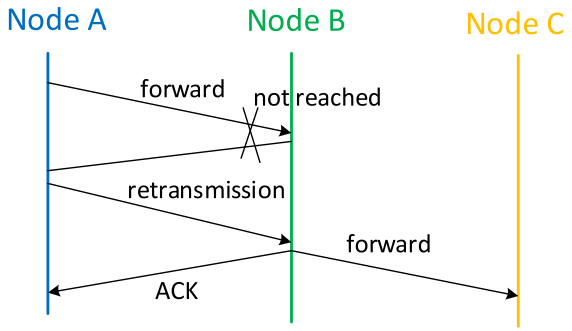

FIGURE 4. Case of transmission failure and retransmission in QELAR.

be updated according to $V_s = \max_a Q(s, a)$. The state space of the algorithm contains the node that holds the packet. This action is represented as packet forwarding by a node. The reward function is designed using the cost function of the sender node's residual energy and the energy distribution among the group nodes. When choosing an action from a state, the Q-values of all actions from that state are calculated first. Then, the action with the maximum Q-value is chosen, and the V-value of the state is updated.

One important feature of QELAR is the acknowledgment (ACK)-receiving mechanism, which confirms packet transmission. The sender node does not remove the packet from the buffer immediately after sending; rather, it waits until the next forwarder forwards the packet to the next-hop node. Thus, retransmission is triggered if the forwarder does not overhear transmission. The transmission failure and retransmission mechanisms are shown in Figure 4.

As we can see from the figure, node A waits till node B forwards the packet to node C. Upon transmission failure, node A retransmits the packet and when node B forwards the packet to node C, it receives that as ACK. However, the number of retransmission is limited by a predefined value max_{trans}.

- **Advantages:** Being one of the earliest Q-learning-based UWSN routing protocols, QELAR designs the RL algorithm to make routing decisions in a way that it could be further improved. The method used here for transmission confirmation after sending a packet reduces the communication overhead and the number of packet drops.
- **Disadvantages:** The reward function is designed considering only the residual energy whilst neglecting other important selection parameters, such as the distance or depth of the neighbor nodes. Sometimes, a node may have more residual energy but a higher distance; hence, the energy consumption will increase when traveling over longer distances.
- **Application:** QELAR is designed for a UWSN environment in which the source node is fixed; meanwhile, the other sensor nodes are dynamic. QELAR is unsuitable for UWSN routing across networks in which all nodes are dynamic, because the source node can be any one of the nodes.

B. MULTI-LEVEL ROUTING FOR ACOUSTIC-OPTICAL HYBRID UWSN (MURAO)

Extending their work in [47], the authors of [48] proposed another routing protocol named MURAO, which was designed for an acoustic–optical hybrid UWSN environments. A multilevel Q-learning method suitable for a multilevel UWSN was applied. The multilevel distributed Q-learning approach accelerates convergence. In this type of approach, the state space is divided into different groups, where each group contains one agent. This agent supervises all other lower-level agents, whilst logically being in the higher level. Thus, the number of states becomes smaller, which accelerates termination. A clustering method is applied in MURAO, in which clusters are updated based on changes in network topology. The routing process consists of several concurrent lower-layer routings inside cluster members and one upper-layer routing among the CHs. Several gateway nodes exist in the network; these connect two clusters because the nodes that receive broadcast messages from multiple CHs become members of multiple clusters and eventually function as gateway nodes.

The inter-cluster routing process in the upper layer is realized via both the acoustic and optical channels, whereas the intra-cluster routing in the lower layer is performed using only the optical one. The CHs in the upper layer assign gateway nodes to the clusters, by applying Q-learning. The gateway nodes represent the destination nodes for each intra-cluster routing assigned by the CHs, which is the terminal state in the Q-learning approach. The intra-cluster routing process is similar to that in QELAR. The Q-values and V-values are updated after each action. Routing is initiated in one of the gateway nodes and terminates when it reaches the designated gateway node. Three types of information exchange occur in the network: (1) between cluster members, (2) between CHs, and (3) between cluster members and CHs.

- **Advantages:** Applying the multilevel Q-learning algorithm to multiple layers of the UWSN accelerates the algorithm convergence. The number of states is reduced by applying the algorithm to different clusters; this also helps the algorithm to reach the terminal state faster. Hybrid communication exploits both acoustic and optical channels.
- **Disadvantages:** Applying Q-learning separately for each cluster can complicate the network. Although it reduces the number of states for each cluster, the computational costs may increase. Moreover, in a dynamic UWSN, the clustering changes according to node mobility, so the routing will also be changed.
- **Application:** MURAO is more suitable for a static UWSN environment. In such scenarios, clustering will occur only once, and routing will be more efficient.

C. Q-LEARNING BASED DELAY-AWARE ROUTING (QDAR)

Jin *et al.* proposed QDAR routing algorithm with an objective to extend the network lifetime of UWSNs [49]. Q-learning was used because it can determine the globally optimal next hop, rather than a greedy one. The routing decision is taken with regard to the propagation delay and residual energy. A multi-agent Q-learning technique is employed by considering each packet in the network as an agent. The sink node performs a virtual experiment, utilizing the algorithm to determine a routing path by sending a virtual packet in the virtual topology, because the sink node possesses information regarding the nodes. The overall routing mechanism is divided into five phases: data ready phase, routing decision, interest phase, packet forwarding, and acknowledgment. A flowchart of the routing mechanism is presented in Figure 5

Three assumptions are considered: the depth information of each node is held by those nodes, and it can be embedded in the packets; nodes implement Source_initiates_Query; and the records of successful or failed communication are saved in the sink node. The source node sends a DATA_READY packet to both request communication and collect information in a reactive manner; hence, the source node must send a packet to the sink node. The neighbor node whose depth is smaller than that of the previous node forwards the packet only. In the routing decision phase, the QDAR algorithm is applied to select the routing path. Through Q-learning, the next-hop node is selected from the neighboring nodes, to optimize the residual energy and propagation delay. After the sink node makes the routing decision, it creates an INTEREST packet in the interest phase; this is sent back to the source node as an acknowledgment.

- **Advantages:** The algorithm mitigates the trade-offs between network lifetime and end-to-end delays in an adaptive and distributive manner. The virtual topology concept used in this protocol increases the cost of failed transmission.
- **Disadvantages:** The packets are considered as the agent in the network, and the state is the node that holds the packet. Under an increasing number of nodes and packets in the routing path, the number of states also increases. This will increase the state space, and the algorithm may fail to converge.
- **Application:** This protocol is suitable for both static and dynamic underwater environments. Therefore, it can be applied to UWSNs in which the dynamic topology changes. It can function in an adaptive and distributive manner.

D. HARVESTING-AWARE DATA ROUTING (HyDRO)

Basagni *et al.* proposed a routing protocol (referred to as HyDRO) in an energy-harvesting UWSN, by assuming all nodes to be capable of energy harvesting [50]. This protocol considers both the residual and harvested energy in its optimization. The sender node acts as an agent of the RL. The action aims to select the forwarding node and thereby the route to the sink. The algorithm considers residual energy, foreseeable harvestable energy, and link quality when choosing the route. All of these optimization criteria are considered when a sender node must select a relay node for forwarding a packet to the sink node. The reward function is designed with

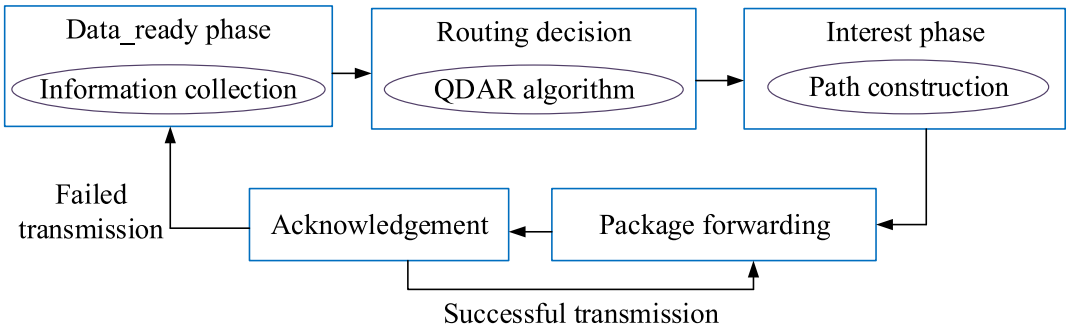

FIGURE 5. Flowchart of routing mechanism in QDAR.

a penalty for packet dropping, to reduce the packet drop ratio. The sender node i always possesses information regarding the neighbor node j to be selected as a relay node. Flooding for route acquisition occurs only during the initiating period and upon returning from an all-off or temporarily malfunctioning state. The all-off state of a node occurs when the node is out of energy. The nodes proactively update their neighbor nodes according to the signal received at a given time. When node i does not receive the signal from node j at time t, it temporarily removes j from its active neighbor list. Node j notifies its neighbors just before running out of battery, by setting a field in its header. Upon transmission failure, the sender retransmits the packet for a given period of time, after which the packet is dropped.

- **Advantages:** The penalty given for packet dropping ensures that the nodes select more reliable relays and route the packets through shorter routes.
- **Disadvantages:** Network lifetime considerations are not considered for performance comparison in this protocol; however, this is an important metric for evaluating a routing protocol.
- **Application:** The protocol is only applicable to an underwater environment in which energy harvesting is possible. It exhibits performance degradation without this harvested energy; thus, it cannot be applied to UWSNs without energy harvesting.

E. Q-LEARNING GAME-THEORETIC DISTRIBUTED ROUTING (QGDR)

In [51], the author proposed a distributed routing protocol for UWSNs, by integrating a game-theory approach with Q-learning (QGDR). The sensor nodes are assumed to be individual agents; they try to maximize their profit by making a cooperative routing decision that is acceptable to all other agents. The nodes learn the policy to select the optimal strategy according to the RL algorithm. The routing problem is designed as a multiplayer routing game model that extends the MDP problem. A new game model is developed following the assumptions for UWSNs, referred to as the routing game. First, the UWSN topology is configured with the help of the configuration algorithm proposed in this study. The topology is formed using a payoff history array $U[.]$ and a path_cost value PC which determines the cost of the link in the source-to-sink-node route. A virtual topology is structured and can be dynamically reconfigured by changing these two values.

- **Advantages:** This protocol can adapt to dynamic topology changes, which is practical for UWSN scenarios. The sensor nodes can adjust their learning parameters according to changes, and they can dynamically take routing decisions.
- **Disadvantages:** This protocol does not consider network lifetime, which is a necessary parameter. For cases of route failure, no retransmission scheme is mentioned, only a penalty. This may increase the initial packet drop rate when the agent learns.
- **Application:** This protocol is applicable to UWSN environments involving node mobility, because it can function under dynamic changes in the environment. In security and military applications where the dynamic environment is necessary, this protocol can be applied to provide information.

F. Q-LEARNING-BASED EFFICIENT AND BALANCED ENERGY CONSUMPTION DATA GATHERING (QL-EEBDG)

Karim *et al.* proposed QL-EEBDG in [52], by considering the void hole problem for routing in a UWSN. This problem occurs when a selected next-hop node does not have any neighbor node or does not lie within range of the sink node. This leads to an increase in packet dropping and energy consumption. To mitigate the void hole problem, only nodes that have a next-hop node are selected as forwarding nodes. Each node functions as the Q-learning agent, where the sender node is the source agent and the neighbor node is the receiver. A control packet is generated from all nodes and sent to the neighbor nodes. Then, the neighbor nodes send back acknowledgment packets by which the sender node declares the neighboring nodes. Then, the Q-value of all neighbor nodes is calculated; the nodes with the highest Q-value represent the shortest distance towards the sink.

Based on the distances of the nodes, three types of rewards are computed: reward sink, for choosing the sink as the next node; reward pos, for choosing a neighbor node; and reward neg, for choosing neither a sink nor neighbor node. The node

with the maximum Q-value is selected as the next-hop node. If more than one node has the same Q-value, then that with a higher residual energy is selected as the next-hop node. A circular network topology is created using a static sink node and sender nodes. Another parameter (MS) is used in the simulation; it moves clockwise. When a node must send data, it determines whether the MS is within the shortest transmission range. Then, the sender node sends the data to the MS; otherwise, it utilizes the Q-learning-based method to choose the next-hop node.

- **Advantages:** In this protocol, only the nodes for which either one neighbor node or sink exists in the one-hop distance are selected as the neighbor nodes. Therefore, even if a node with no further one-hop node is nearer than the source node, it will not be selected as a neighbor. This procedure helps to reduce the void hole problem, leading to fewer packet drops and lower energy consumption.
- **Disadvantages:** No retransmission strategy or penalty is applied in the Q-learning algorithm upon route failure. This may lead to an increased packet-drop rate. Moreover, the agent (sensor node) does not consider the end-to-end delay when choosing the next-hop node.
- **Application:** This routing protocol can be applied to a dynamic UWSN environment because the neighbor nodes can be selected dynamically. If the UWSN exhibits node mobility, this algorithm can be utilized to select the next-hop node.

G. Q-LEARNING BASED LOCALIZATION-FREE ROUTING (QLFR)

Zhou *et al.* [53] proposed the routing algorithm QLFR for UWSNs; their objective was to extend the network lifetime and minimize the end-to-end delay. When a node has to send a packet, it checks the Q-values of all neighboring nodes and places these nodes in a priority list sorted in the reverse order of Q-values. The nodes with a smaller depth will have a higher priority. The priority list is added to the data packet sent to the neighboring nodes. The nodes in the list hold the packet, following a holding time mechanism provided by Q-learning. The other nodes drop the packet. The holding time mechanism design is shown in Figure 6. Here, when node s wants to send a data packet, the three neighboring nodes p, q and r will receive it. The depth of r is lower than that of the sender; therefore, it will drop the data packet. For the remaining two nodes, if p has a higher Q-value than q, it is selected as the next-hop node. The Q-value is calculated according to two cost functions: depth-based cost and energy-based cost. Therefore, the reward is designed considering these two parameters. When node s_i sends a packet to s_j following the action a_j, the reward can be calculated by

$$r_{s_i s_j}^{a_j} = -c_e(s_i) - c_e(s_j) - c_d(s_i, s_j), \tag{5}$$

where $c_e(.)$ is the energy-based reward and $c_d(.)$ is the depth-based reward; both lie within the range of [0,1].

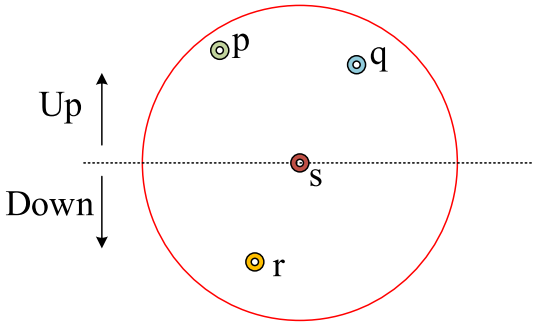

FIGURE 6. Graphical representation of holding time mechanism in QLFR.

Furthermore, a packet-delivery ratio-based multipath-suppression mechanism was proposed to maintain the priority list length. The packet delivery ratio was calculated to control the length of the priority list for reducing unnecessary transmission.

- **Advantages:** The holding time mechanism causes the nodes with a lower depth to drop the packet; this in turn reduces the redundant retransmission in the network. Moreover, the overall holding time of the packet is reduced because the node with the highest Q-value transits without holding.
- **Disadvantages:** The nodes not included in the priority list drop the packet. This increases the packet drop ratio in the network; in particular, when the node density increases, more nodes are amongst the neighboring nodes of the sender node but have lower depths; hence, more nodes will drop the packet.
- **Application:** This protocol is suitable for UWSN environments with underwater monitoring applications and few nodes. Basically in the application, when the source node is anchored to the ocean floor and transmits data to the upper nodes, toward the sink. So, data are only passed from nodes with more depth to nodes which have less depth in the underwater environment.

H. CHANNEL-AWARE RL-BASED MULTI-PATH ADAPTIVE ROUTING (CARMA)

Valerio *et al.* proposed the routing protocol CARMA in [54], to select the set of relay nodes in a UWSN. Their main objective was to simultaneously optimize the route-long energy cost and maintain the network lifetime and packet delivery ratio. The size and composition of the relay set are determined dynamically at each transmission time. When a node sends a packet to the sink node, it discards the packet and transmits it to all nodes in the list of relay nodes. All required information is added to the header of the packet. An implicit acknowledgment mechanism is used to overhear the retransmission of the packet. The sender node waits for this acknowledgment for a specified period. If no acknowledgment is received, the packet is retransmitted thereafter. The transmission is considered successful when acknowledgment is received. During the entire procedure, RL is employed to select a list of relay nodes when forwarding the packets. The RL agent selects

this list according to the local channel quality and energy consumption across the entire route.

Initially, the nodes have no knowledge of the environment and with experience, the nodes learn and update their knowledge. When a node sends a packet to the sink node, it uses the Q-value to obtain the optimal route. This algorithm chooses the optimal action according to the value of the action, which is the cost required to transmit the packet from the sender to the sink node. Furthermore, increasing numbers of retransmissions affect the network performance for increased network traffic. Therefore, the maximum number of retransmissions is determined dynamically by utilizing the well-known ALOHA closed-form expression, $S = Ge^{-2G}$. Here, G is the average number of transmission attempts in a time interval equal to that required to transmit one packet.

- **Advantages:** The size and composition of the relay set at each transmission attempt is determined dynamically, which increases the packet delivery ratio and ensures a lower energy cost. Another useful feature of this protocol is that it facilitates packet forwarding, by broadcasting a packet when no neighbor node is known to the sender.
- **Disadvantages:** CARMA considers only the static UWSN environment, which may not be suitable for all types of USWN scenarios in which the node exhibits mobility. Moreover, it selects a set of relaynodes (instead of the one-hop neighbor relay) at a time, which may lead to network performance degradation in cases of higher network traffic.
- **Application:** This protocol is suitable for a static UWSN environment (i.e., where nodes are deployed in stationary positions and when only the data from that position are delivered to the sink node). This routing protocol is suitable for monitoring temperature and other environmental attributes.

I. RL-BASED CONGESTION-AVOIDED ROUTING (RCAR)

In [55], Jin *et al.* investigated the congestion control problem in UWSN routing, and they proposed an RCAR protocol to minimize energy consumption and end-to-end delay. The protocol comprises four stages: initialization, virtual pipe creation, virtual routing, and packet forwarding. In the initialization stage, all nodes exchange their location information and residual energy with one-hop neighbors collected from the physical layer. A neighbor table is generated in each node using the one-hop neighbor information. Then, a dynamic virtual routing pipe is generated by the node that holds the packet forward. The radius of the pipe is based upon the average residual energy of the neighboring nodes. The radius of the pipe is given as

$$R^i_{\text{pipe}} = -\frac{2(R - R^i_{\text{ini}})}{E_{\text{ini}}} \times \overline{E^i} + 2R - R^i_{\text{ini}}, \tag{6}$$

where E_{ini} and R are the initial energy and transmission range of the nodes, respectively. R^i_{ini} is the initial radius of the pipe,

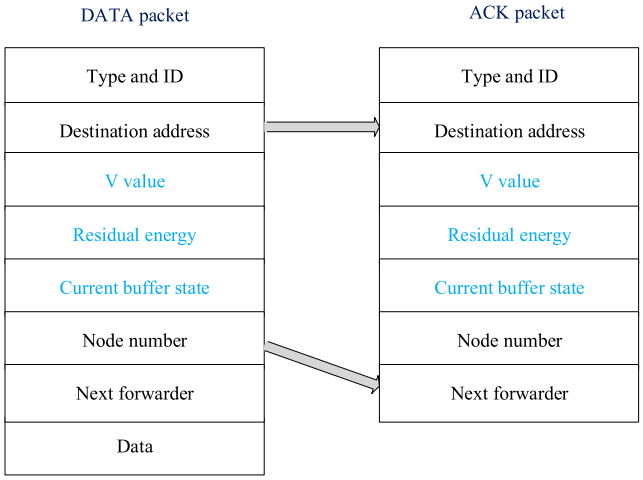

FIGURE 7. Structure of DATA packet and ACK packet in RCAR.

and $\overline{E^i}$ is the average residual energy of the next-hop nodes. This virtual pipe helps to reduce unnecessary initial exploration detours. Then, virtual routing is performed using the RL-based algorithm to select the next forwarding node. After the node forwards the packet, packet forwarding is considered completed if the selected node is available. If unavailable, the algorithm is reapplied to choose the next-hop node, and the information regarding the link is updated. A handshake mechanism based on S-FAMA is utilized in the MAC layer to update the node information in the initially generated neighbor table. During this period, DATA and ACK packets are used to exchange information. In RCAR, three additional pieces of information are included in these packets: residual energy, current buffer state, and V value. These determines the value of a node for selection as a next-hop node. The structures of the DATA and ACK packets are shown in Figure 7, with the additional information highlighted. When a node has to send a packet again, the Q-value is calculated using the updated information.

- **Advantages:** Unlike other RL-based UWSN routing protocols, RCAR prevents congestion in the network. The handshake-based method for updating information in the MAC layer helps to reduce energy consumption, because nodes need not broadcast periodically to update their information. It also mitigates collisions between nodes during broadcasting.
- **Disadvantages:** The state-space contains information regarding the one-hop neighbors of each node. Under an increase in the number of nodes in the environment, the number of one-hop neighbors also increases. In this case, the state size becomes large, and the algorithm takes more time to converge.
- **Application:** This protocol can be applied to dynamic underwater acoustic communication-based networks, because it can adapt to dynamic topologies. It is suitable for UWSNs, where the number of sensor nodes is moderate.

J. Q-LEARNING BASED ENERGY-DELAY ROUTING (QL-EDR)

Wang *et al.* proposed a clustering-based routing protocol QL-EDR in [56], employing Q-learning learning to select the next-hop node in a hierarchical UWSN. Communication between the nodes was considered via magnetic induction. The main objective of this protocol is to extend the network lifetime by minimizing energy consumption and end-to-end delay. The framework is divided into three parts: data collection, data processing, and decision management. In the first phase, cluster heads (CHs) are selected by forming several clusters in each layer of the three-layered UWSN. Cluster members send their sensed data to the CHs for transmission to the base station. In the second phase, the data features from the data sent by the sensor nodes are extracted. The third phase employs the Q-learning algorithm to select the next one-hop node according to the residual energy and distance of the nodes. Two parameters are used to obtain a single-hop bonus, from which the reward function and Q-values are designed. The parameters are D_{hop} for the distance-based bonus and E_{hop} for the residual energy-based bonus. They can be calculated by

$$D_{hop} = \frac{D_t}{d_t + D_{t+1}}, \tag{7}$$

$$E_{hop} = \frac{E_t}{\frac{1}{e_t} + E_{t+1}}, \tag{8}$$

where D_t is the shortest-distance-based path, d_t is the distance between two nodes, E_t is the maximum energy-based path, and e_t is the residual energy of the nodes. A regulatory factor β is used to emphasize the value of the residual energy or transmission delay according to the state, to prolong the network lifetime.

- **Advantages:** The clustering of sensor nodes helps to minimize the overall end-to-end delay and energy consumption in the network, because not all sensor nodes need to join as the relay node, and only the CHs will perform data transmission.
- **Disadvantages:** The multi-hop path is selected after the CH receives all the data in the cluster. If a single node (rather than all the nodes in a cluster) must send data, the node must wait until all nodes send data to the CH. This increases the latency for a single sensor node, even if it decreases the overall end-to-end network delay. Moreover, if a cluster member is just one hop away from the sink, it cannot transmit data, because it must send data to the CH. CH selection was not optimized here.
- **Application:** QL-EDR is not suitable for emergency applications in UWSNs because data are sent by CHs. However, it is applicable in environments where sensor nodes only perform monitoring tasks after a specific timestamp. All nodes send data to the CH at that time, and the CH transmits it to the sink node and eventually the base station.

K. DISTRIBUTED MULTI-AGENT RL ROUTING (DMARL)

Li *et al.* proposed DMARL in [57] as a routing protocol for UWSNs, by considering an optical communication medium. They designed the UWSN as a distributed multi-agent system that supports information interaction between adjacent nodes. Subsequently, a multi-agent RL algorithm is applied in the routing process, to prolong the network lifetime and adapt to the dynamic topology of the UWSN. The implementation of DMARL is performed in three stages: preliminary stage, route discovery, and route forwarding. The preliminary stage involves the sensor node deployment and routing-table-related parameter initialization. Then, in the route discovery stage, each node determines its one-hop neighbors by periodically broadcasting hello packets. The Q-table is updated according to the neighbor node information. Q-learning is applied in the route-forwarding stage. Each node operates as an agent and maintains a Q-table. The state is regarded as a node with a data packet at a particular time t. The action of the agent is to select the next-hop node. The reward mechanism is designed based on local and global rewards. The local reward function is designed considering the residual energy and link quality between sensor nodes; these are received by an ACK packet after data transmission. The local reward can be defined as:

$$r^{t+1}(s_j^{t+1}) = \begin{cases} K_{\text{non-ACK}}, & \text{w/o ACK} \\ W_E.E + W_L.L_Q, & \text{w/ ACK} \end{cases}, \tag{9}$$

where E is the normalized residual energy of the receiver node j, and L_Q is the normalized link quality. W_E and W_L represent the weights of the residual energy of node j and link quality, respectively. $K_{\text{non-ACK}}$ represents a negative reward when no ACK is received (i.e., when the routing quality is poor). The global reward is given here to obtain feedback regarding changes in the environment; this reward depends on the transmission direction—that is, upon whether the current node is closer to or farther from the sink node than the previous node. If it is closer, then a positive reward is given; otherwise, a negative reward is given. One important aspect of DMARL is that, to accelerate the convergence of the RL algorithm, two optimization strategies are utilized: position-based Q-value initialization and learning-rate variation. The first strategy initializes the Q-value according to the initial distance to the sink from a node and one of its neighbor nodes. In the second strategy, the learning rate is optimized according to link stability, to reflect the changes in the neighbor set.

- **Advantages:** Q-value initialization helps to decrease the learning time of the RL algorithm. Moreover, adjusting the learning rate according to the dynamic environment accelerates the algorithm's convergence. Integrating both techniques reduces the number of training steps and accelerates convergence, which subsequently saves energy in UWSNs.
- **Disadvantages:** DMARL is specifically designed for underwater optical communication. In addition,

as mentioned in the paper, DMARL is unsuitable for a UWSN environment in which more than 14 neighboring nodes (on average) are present. Despite its good performance in dynamic environments, the node density constraint limits its performance to specific UWSN environments.

- **Application:** DMARL is designed considering node mobility in a UWSN. Therefore, this protocol can be applied to any dynamic UWSN environment featuring a limited number of nodes.

L. ENERGY-EFFICIENT DEPTH-BASED OPPORTUNISTIC ROUTING WITH Q-LEARNING (EDORQ)

Lu *et al.* proposed a routing protocol EDORQ [58] for UWSN; they sought to ensure energy-saving and reliable data transmission from sensor nodes to sinks. The overall routing process of EDORQ consists of two stages. First, candidate-set selection is performed based on void detection; second, candidate-set coordination is performed via Q-learning. The first stage aims at choosing candidate nodes from the neighboring nodes to forward the packet to the sink. The depth and void-flag information of the nodes are used as candidate-selection metrics. The candidate-set selection is composed of two modes: a greedy mode and void-recovery mode. In the greedy mode, the nodes closer to the sink than the sender node (according to depth) are selected as candidate nodes. A void can arise when a forwarder node is selected (because of its smaller depth) but has no further forwarder node closer to the sink. Void recovery is triggered when a node returns the packet upon detecting a void and does not receive information that the packet has been successfully forwarded. In such a case, the node for which a void is detected selects a forwarder node with a greater depth.

The candidate set coordination stage utilizes the Q-learning algorithm along with a holding time mechanism to select the forwarder node. The reward function in the Q-learning algorithm is designed with residual-energy-related values, depth-related values, and void-detection factors. Using this reward function, the Q-values of all nodes in the candidate set are calculated. Subsequently, a holding time is assigned to all candidate nodes, such that the node with a higher Q-value has a lower holding time and therefore a higher priority for sending the packet. The mechanism is illustrated in Figure 8. Nodes n_2 and n_3 are in the candidate forwarder set for sender node n_1. The holding time is calculated for both nodes according to the Q-values of nodes n_2 and n_3. In the figure, node n_2 forwards the packet when the timer is over and, upon overhearing the transmission, node n_3 discards the packet. The holding time of node n_j is defined as follows

$$T_j = [1 - \frac{2}{\pi} \arctan Q(s_i, a_j)]T_{\max}, \tag{10}$$

where $Q(s_i, a_j)$ is the Q-value of node n_j in state s_i, and $T_{\max}$ is the predefined maximum holding time calculated using the maximum communication range of a node and the propagation speed. This holding time is used to prevent the overheads entailed by packet forwarding of all candidate nodes. When a candidate node with lower priority overhears the same packet transmission from a higher priority node within its holding time, it drops the packet. Thus, the optimal candidate node is selected for packet forwarding.

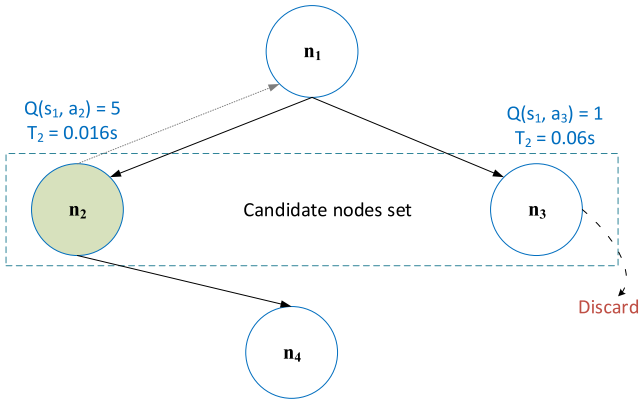

FIGURE 8. Candidate set coordination mechanism with Q-learning and holding time in EDORQ.

- **Advantages:** The reward design ensures that, upon selecting a node with higher residual energy, smaller depth, and greater void-detection factor, the agent receives a higher reward. The candidate set coordination does not require any additional ACK packet transmission, owing to the holding time mechanism. Moreover, this protocol ensures reliable packet transmission because each node holds the packet until the end of its holding time before dropping it.
- **Disadvantages:** The holding time mechanism increases the delay because each node must wait until the end of its holding time before the packet is forwarded. This increases the end-to-end delay for routing in the network.
- **Application:** EDORQ is suitable for dynamic topologies, because it is an on-demand routing protocol that can be adjusted according to node mobility. However, this protocol may be unsuitable for time-critical applications, because each forwarder node must wait until the end of the holding time, which may cause a delay.

M. REINFORCEMENT LEARNING-BASED OPPORTUNISTIC ROUTING (RLOR)

Zhang *et al.* proposed an opportunistic routing protocol with RL (named RLOR) for UWSN acoustic communication [59]. In an opportunistic routing procedure, packet routing is performed via the cooperation of multiple nodes receiving the packet rather than a single relay node. When a node n_i must send a data packet, the candidate forwarding set of the node is determined first. This set of candidate nodes is selected according to the depth, energy, and number of neighbor nodes. Then, the sender node sends its location information to its neighbors in the candidate set. The next-hop node from the candidate nodes is selected using an RL algorithm. In this

RL algorithm, the state space contains information about the current node, as well as its candidate node set. The action is to select the next-hop node from the candidate node set. The reward for taking action a_j (of selecting the next-hop node n_j from node n_i) can be defined as

$$R_{n_i,n_j} = \Delta_{\text{dep}(i,j)} \cdot p(d,l) \cdot G_{\text{above}}(n_j) \cdot E(n_j), \qquad (11)$$

where Δ_{dep} is the depth difference between nodes i and j, $p(d,l)$ is the probability of successful packet transmission, $G_{\text{above}}(n_j)$ is the number of neighbor nodes above n_j, and $E(n_j)$ is the energy of n_j. Upon receiving a packet transmission request from the sender node, all candidate nodes calculate their Q-value using the discounted reward function. After receiving the Q-values of the candidate nodes, the sender node selects the next-hop node with the largest Q-value. In the case of a routing void problem (i.e., when selecting a node with no neighbor nodes to transmit the packet to), a method called the recovery mode is applied. When a void is detected, the void node enters the recovery mode and selects a downward forwarder node using the RL algorithm. The forwarder node is selected based on the smaller depth differences and higher energies. In addition, opportunistic routing is used alongside an adaptive dynamic timer mechanism. A waiting time is set for every candidate node (according the communication delay), to choose the forwarder node with higher priority. The node for which the waiting time elapses first forwards the packet, and other candidate nodes will drop it.

- **Advantages:** The adaptive dynamic timer mechanism ensures successful packet transmission. In addition, the waiting time of the candidate nodes leads to the selection of the best relay node. The end-to-end delay is also reduced by considering the communication delay as a parameter for setting the waiting time.
- **Disadvantages:** In RLOR, the state space contains the current node and set of candidate forwarder nodes. In a dynamic UWSN, the candidate set varies during the routing process at different times. Therefore, the state space is larger, and the RL algorithm may be slow to reach convergence.
- **Application:** RLOR exhibits better performance in the case of a dense network, because the risk of routing void problems is smaller. Moreover, this protocol can only be applied to UWSNs operating via acoustic communication.

V. COMPARATIVE STUDY AMONG ROUTING PROTOCOLS

In this section, a comparative study of all the investigated routing protocols is presented from different perspectives. The key ideas of all routing protocols are listed in Table 3. The specific features of each routing protocol used to select forwarding nodes or routing paths differ, despite all being based upon RL. The main ideas of the protocols are compared and discussed in Section V-A.

A. KEY IDEAS OF RL-BASED ROUTING PROTOCOLS

The novelty of QELAR novelty lies in the design of its reward function, which contains two cost functions related to the residual energy: one relates to the residual energy of the node holding the packet, the other relates to the energy distribution in the group of direct neighbors of that node. The average residual energy of the neighboring nodes is thereby computed. The key idea of MURAO is to physically divide the network into several clusters and logically partition these clusters into two layers. Multi-level RL is applied to both layers, and the upper layer contains the cluster heads that function as agents for their associated clusters.

In QDAR, the sink node implements Q-learning to determine the routing path; meanwhile, each packet functions as an agent. Using all information from the UWSN nodes, the sink nodes create a virtual topology by sending a virtual packet. However, the HyDRO protocol selects the route that maximizes the residual energy; hence, when choosing the neighboring node to the sink node, the main objective is to select the nodes with maximum residual energy. The reward is designed such that the relay node selected for sending packets always as the maximum residual energy.

QGDR assumes that each sensor node is a player in a multiplayer routing game. Each node learns the policy of choosing the best relay node to send packets to the sink node, by utilizing Q-learning. QL-EEBDG aims to mitigate the void hole problem encountered in UWSNs by selecting a node with at least one neighbor node or the sink within their one-hop distance. Then, the node with the shortest distance is selected as the next-hop node.

In QLFR, a new holding-time mechanism is designed using RL, to schedule packet forwarding according to node priorities. This mechanism helps reduce redundant transmissions between multiple forwarding nodes. Unlike other RL-based routing protocols, CARMA chooses a set of relay nodes to forward packets toward the sink node, using the RL algorithm. However, most protocols choose the next-hop neighbor only. The channel condition and route-long energy are considered when determining the list of relay nodes.

The RCAR is performed by each node that holds a packet forward. The node creates a dynamic virtual routing pipe using the residual energy of the neighboring nodes, and it performs virtual routing to select the next forwarding node. A clustering approach is used in the QL-EDR to collect data from the sensor nodes. After all data in the cluster are collected by the CH, Q-learning is adopted to select the next hop. The residual energy and transmission delay are used as indicators for selecting the routing path.

B. COMPARISON OF THE RL APPLICATIONS

RL has been applied to different protocols to solve different problems. In all investigated protocols, the designs of the RL algorithm differed. The state, action, and reward functions were constructed with different objectives. The reviewed protocols are compared in terms of RL applications in Table 4.

TABLE 3. Key ideas of RL-based UWSN routing protocols.

Protocol	Ref.	Key idea
QELAR	[47]	The energy consumption of nodes and residual energy distribution between nodes are considered for learning of Q-learning algorithm.
MURAO	[48]	The network is divided into multiple layers with acoustic-optical hybrid communication, and Q-learning is applied in a multi-level distributed manner.
QDAR	[49]	The sink node uses a Q-learning algorithm to determine a routing path by sending a virtual packet in the virtual topology.
HyDRO	[50]	The sender node selects a route for sending packets that maximizes the residual energy on the entire route toward the sink node.
QGDR	[51]	Each sensor node is assumed to be a player in a multiplayer routing game with the objective of finding the best routing policy.
QL-EEBDG	[52]	A node is considered to be eligible as a forwarder node only if a next-hop neighbor is present to mitigate the void-hole problem.
QLFR	[53]	The holding time mechanism is used to schedule packet forwarding, which reduces redundant transmission between multiple forwarding nodes.
CARMA	[54]	The sender node executes RL to determine the set of next-hop relays based on channel conditions and route-long costs.
RCAR	[55]	Each sender node performs virtual routing using RL to decide the next forwarder.
QL-EDR	[56]	Data collection is performed by clustering, and the next-hop node is selected by Q-learning using the residual energy and delay.
DMARL	[57]	Multiagent RL is applied with a distributed reward strategy to exploit the information interaction between adjacent nodes.
EDORQ	[58]	A candidate set with potential forwarder nodes is selected, and Q-learning is applied with a holding-time mechanism to select the forwarder node.
RLOR	[59]	The RL algorithm is applied with opportunistic routing to select the best forwarder node as well as a recovery mode for the routing void problem.

In most protocols, the RL algorithm is designed in a distributed manner, where each node acts as the RL agent. The sender node observes the states (essentially the Q-value of neighboring nodes) and chooses either the next-hop node or the routing path. Some RL designs are also centralized such that the sink node functions as the agent and chooses the routing path. Here, the RL designs in the reviewed protocols are described individually.

As shown in Table 4, in all the reviewed protocols (except for MURAO, QDAR, and QL- EDR), the sensor nodes function as the agent of the RL algorithm. In general, the source node or sender node becomes the agent and performs data forwarding by following a policy or according to the maximum Q-value. In MURAO, routing is performed in a hierarchical manner in which the CHs select the routing path in the corresponding clusters and perform packet routing. However, QDAR assumes that each packet is an agent of the Q-learning algorithm. The agent's policy is the routing path that directs the packet (agent) to take proper actions. In QL-EDR, the base station observes and makes routing decisions as the Q-learning agent.

The agent's state is the observation factor of the RL agent, from which the agent decides the action. The information available in the state is crucial for the agent learning process. In addition, the state space should not become large. In QELAR, MURAO, QDAR, DMARL, and EDORQ, the state of the algorithm relates to the node that holds the packet. Therefore, at any time t, the state in the RL algorithm is the ID of the node where the packet resides at that time. The routing action is selected according to the ID of that node. Retaining the nodes holding the packet as the state is beneficial, because the next state will be the next-hop node. Therefore, consecutive states form the routing path for the routing protocol.

The state spaces in HyDRO and CARMA are similar. They contain sets indicating the number of times node i has transmitted packet p unsuccessfully, as well as the packet transmission or packet drop. Depending on the status of the packet (i.e., whether it is received or dropped by the neighbor nodes), the packet status is established in the state of the node. The state is designed according to single-packet forwarding. In QGDR, the goal of RL is to identify the optimal routing policy in which each node is considered as a player in a multiplayer MDP problem. The state consists of a payoff history array and path cost value. The reason for designing such a state-space is to transmit the packet to the sink with the maximum payoff.

The state space in QL-EEBDG includes the control packets generated by the source node; meanwhile, the source node is referred to as the source agent. These control packets are sent by the source nodes to all sensor nodes within its range. The receiver node, as the receiver agent, sends back an ACK packet, which is used to select the neighbor node. In QLFR, the residual energy and depth of the node comprises the state of each node. With these two types of information, the selection of the next-hop node has the advantage of selecting the optimal forwarding node. The state of a node in RCAR is designed with information regarding one-hop neighbors and the link condition between them. The Q-value of each node is calculated from this information, and the highest Q-value

TABLE 4. Applications of RL in the reviewed routing protocols.

Protocol	Agent	States	Action	Reward
QELAR	Source node	Node holding the packet	Packet forwarding	Residual energy, energy distribution of nodes
MURAO	Cluster head	Node holding the packet	Packet forwarding to neighbor node	Residual energy, energy distribution of nodes
QDAR	Packet	Node holding the packet	Select next-hop node	Cost for successful and failed transmission
HyDRO	Sender node	Number of packet transmission and packet drops	Forwarding packet to neighbor	Function to maximize residual energy
QGDR	Sensor node	Payoff history array and path_cost value	One-hop routing	Payoff to the sender node
QL-EEBDG	Sensor node	Control packet	Select next-hop node	Distance towards sink
QLFR	Sensor node	Residual energy, depth	Select next-hop node	Cost based on energy and depth
CARMA	Sender node	Number of packet transmission and packet status	Select set of relay nodes	Cost for packet dropping
RCAR	Sensor node	Information of one-hop neighbor and link condition	Packet forwarding	Cost for constant, congestion, delay and energy
QL-EDR	Base station	Node information and topological relationship	Select next-hop node	Cost for residual energy and delay
DMARL	Sensor node	Sender node information	Select next hop node	Residual energy, link quality
EDORQ	Sensor node	Node holding the packet	Select next hop node	Residual energy, depth, void detection factor
RLOR	Sensor node	Current node, set of candidate nodes	Select next hop node	Depth, energy, number of neighbor nodes, probability of successful transmission

determines the optimal link condition. The next-hop node can be selected from the optimal link quality in this manner.

The action of all reviewed RL-based routing protocols for UWSNs is the selection of one or more relay nodes for packet forwarding. CARMA acts to select the set of relay nodes; meanwhile, all other protocols act to select the next-hop neighbor.

The reward function in each of the reviewed RL-based routing protocols reflects the main objective of the protocols. For example, QELAR, MURAO, HyDRO, QL-EDR, DMARL, and EDORQ consider the residual energy of the neighbor nodes when designing the reward function, such that the node with higher residual energy can provide higher rewards and therefore be selected as the forwarding node. However, the residual energy alone cannot determine the likelihood of a node being selected as the optimal next-hop node. Therefore, other parameters (e.g., energy distribution among neighbor nodes, link quality, distance or depth of node, and delay) have been added alongside the residual energy. Protocols besides those mentioned above involve reward functions that neglect residual energy, similarly reflecting their objectives.

C. COMPARISON OF THE OPTIMIZATION PARAMETERS

The reviewed RL-based UWSN routing protocols were designed to optimize the performance from different perspectives. Given that the optimization criteria have trade-offs, a routing protocol should attempt to maximize the outcome of the expected performance metrics whilst also minimizing the negative impacts upon other performance metrics. In Table 5, the optimization parameters and evaluation metrics of the reviewed routing protocols are presented. In the table, 'O' indicates that the specific parameter is considered in the specific protocol, and 'X' indicates that the parameter is not taken into account for that protocol.

The residual energy of a sensor node is the remaining energy of the node [60]. This is an important optimization parameter because it determines how long a node can participate in packet forwarding. The residual energy can be computed as

$$E_r = \sum_{i=1}^{n} R_i, \tag{12}$$

where R_i denotes the residual energy of the i-th sensor nodes, and n denotes the total number of sensor nodes. When designing a routing protocol, the next-hop node must be selected by considering its residual energy. All the reviewed protocols (excluding CARMA) optimize the residual energy when making routing decisions. Residual energy can be saved by reducing the energy consumption of the sensor nodes during sensing and data transmission.

The network lifetime also depends upon the energy consumption of the sensor nodes. Therefore, to be efficient, one of the most crucial features of a routing protocol is minimizing energy consumption and thereby extending the network lifetime. Considering and evaluating the network lifetime is essential when designing a routing protocol. The network lifetime can be estimated from the data transmission duration

TABLE 5. Comparison of optimization parameters in the reviewed routing protocols.

Parameters	Residual energy	End-to-end delay	Network lifetime	Link quality	Harvested energy	Packet drops	Dynamic topology changes	Hop count	Sensor node depth	Packet delivery ratio	Congestion control	Node mobility
QELAR	O	O	O	X	X	X	O	X	X	O	X	O
MURAO	O	O	O	X	X	X	O	X	X	O	X	O
QDAR	O	O	O	X	X	X	O	X	O	X	X	X
HyDRO	O	O	X	O	O	O	X	X	O	X	X	X
QGDR	O	O	X	O	X	O	O	O	X	X	X	X
QL-EEBDG	O	X	O	X	X	O	O	O	X	X	X	X
QLFR	O	O	O	X	X	X	X	X	O	O	X	O
CARMA	X	O	O	O	X	O	X	X	X	O	X	X
RCAR	O	O	X	O	X	X	O	X	X	O	O	X
QL-EDR	O	O	O	O	X	X	X	X	O	X	X	O
DMARL	O	X	O	O	X	X	O	X	X	X	X	O
EDORQ	O	O	X	X	X	X	O	X	O	O	X	O
RLOR	O	O	O	X	X	O	X	X	O	O	X	O

(or round) until all nodes are alive or the first sensor node in the network dies [60]. Of all the reviewed protocols, HyDRO, QGDR, RCAR, and EDORQ do not explicitly consider the network lifetime when designing routing protocols.

End-to-end delay describes the average time required for a packet to traverse from the source node to the destination one; this includes the transmission delay, holding time, processing time, propagation delay, and receiving time [61]. While choosing a routing path, it is important to choose the path that will minimize the time required to deliver the packet to the sink node. In a UWSN, the sink node is the destination node. Therefore, to transmit the packet to the sink node faster, the routing protocol must consider the end-to-end delay. However, QL-EEBDG and DMARL have not considered the data-transmission delay from the source node to the sink node when designing their protocols.

The link quality is an important parameter when selecting the subsequent forwarding node, to ensure that a more reliable link is chosen from amongst the candidate nodes. Owing to the highly error-prone nature of underwater wireless links, data transmission over poor-quality links leads to packet losses, which may necessitate retransmission. Because data retransmission increases energy consumption and delay, it is necessary to select a reliable, good-quality link to reduce the likelihood of packet losses [62]. However, the majority of the reviewed protocols do not consider the link quality when designing routing protocols, as can be seen in Table 5.

Of all reviewed routing protocols, only HyDRO considered energy-harvesting-enabled UWSNs. Nodes deployed at different depths harvest energy from the environment to support their operations. In such networks, the nodes at the seafloor harvest energy through turbines; meanwhile, harvesting in nodes closer to the sea surface happen using solar panels attached to floating devices cabled to the nodes. The energy-harvesting-aware protocol can effectively bypass the energy constraints of sensor nodes in UWSNs.

The packet drop ratio is the ratio between the number of packets dropped by the sensor nodes and the total number of packets sent by the source nodes during a data transmission round [63]. Packet drop ratio can be calculated as

$$P_D = 1 - \frac{P_r}{P_s}, \tag{13}$$

where P_r denotes the packets received, and P_s denotes the packets sent during any specified round. Considering the number of packet drops in the routing protocol design ensures proper selection of the next forwarder, to realize successful packet delivery to the sink. HyDRO and CARMA considered the number of packet drops in the state of the agent. Moreover, QGDR, QL-EEBDG, and RLOR also consider this parameter in their routing protocols.

The constant node mobility in the UWSN environment leads to continuous topology changes [24]. Therefore, it is important to consider dynamic topology changes when designing a routing protocol, to reflect real-world UWSNs. However, HyDRO, QLFR, CARMA, QL-EDR, and RLOR do not consider this parameter. The performance evaluation of a protocol does not ensure accurate results if dynamic topology is not considered. Nevertheless, node mobility has also been neglected when evaluating certain reviewed protocols, such as QDAR, HyDRO, QGDR, QL-EEBDG, CARMA, and RCAR.

During the routing process, a packet may have to hop through multiple nodes from the source node to the sink node. Reducing the number of hops can reduce the delay and energy consumption of the overall network. In this regard, choosing the routing path such that the number of hops is lower can improve efficiency. Among the reviewed protocols, hop count is considered in only two: QGDR and QL-EEBDG.

The distance between two nodes is an important parameter for choosing the next-hop node in terrestrial networks. In contrast, in UWSNs, the depth of the sensor nodes also plays a crucial role. Because the sink is placed on the surface, the

TABLE 6. Comparison of the network design and performance evaluation techniques in the reviewed protocols.

Parameters	Deployment	Simulator	Sink	Number of sensor nodes	Number of sink nodes	Data packet size	Simulation time	Compared protocols
QELAR	3D	NS-2 aqua-sim	Single	125, 250	1	NG	NG	VBF
MURAO	2D	OMNET++	NG	256	NG	NG	NG	Flat Q-learning
QDAR	3D	NG	Multiple	80	5	300B	1000s	QELAR, VBF
HyDRO	2D	SUNSET	Single	20, 40	1	100B	6 days	CARP, QELAR
QGDR	3D	MATLAB	Multiple	500	5	70B	100 times	VAPR, GDAR, ELAR
QL-EEBDG	Circular	NG	Single	0.04 per sqm	1	NG	NG	EBDG, EEBDG
QLFR	3D	NG	Multiple	100-500	NG	NG	NG	DBR, EEDBR
CARMA	Rectangular	SUNSET	Single	6,20,40	1	1000 bytes	NG	CARP, QELAR, Eflood
RCAR	3D	NG	Single	100-300	1	50 bytes	2000s	QELAR, HHVBF, GEDAR
QL-EDR	3D	NG	Multiple	117	NG	1 MB	NG	AVN-AHH-VBF
DMARL	2D	NG	Single	25	1	4000 bits	400s	MARL. EMARL, double Q-learning, AODV
EDORQ	3D	NS-2	Multiple	200-800	4	NG	800s	VBF, DBR, QELAR
RLOR	3D	NG	Single	50-600	1	50 bytes	5000s	EE-DBR, VAPR, MU-RAO, RDBF, Flooding

routing path is oriented toward shorter depths. In this regard, the routing process in QDAR, HyDRO, QLFR, QL-EDR, EDORQ, and RLOR is designed considering the depths of the sensor nodes.

The packet delivery ratio is defined as the total number of packets sent until the end of a transmission round. When evaluating a routing protocol, it is necessary to measure its performance in terms of the packet delivery ratio. A higher packet delivery ratio reflects the higher efficiency of the routing protocol. The majority of the reviewed protocols considered these parameters, as shown in Table 5. During the routing of a packet, congestion can be detected because of the flooding for route discovery or route acquisition. Congestion can degrade network performance if left uncontrolled. However, among the reviewed routing protocols, only RCAR is designed to prevent congestion.

D. COMPARISON OF NETWORK DESIGN AND PERFORMANCE EVALUATION TECHNIQUES

The UWSN designs adopted in the reviewed routing protocols differ. The communication channel, deployment, topology, number of nodes, and other parameters are considered in various ways. Moreover, different researchers have used different evaluation techniques; these techniques are summarized in this subsection and presented in Table 6.

In the table, 'NG' indicates that no option is specified in the paper for the routing protocol. The deployment in the table indicates the dimensions of the topology considered for the simulation. Several protocols considered the 3D deployment of UWSNs, whereas others considered 2D deployment. Other geometric shapes (e.g., circular shapes in QL-EEBDG and rectangular shapes in CARMA) have been considered for designing protocols.

The simulation area refers to the width and height of the network scenario in the 2D network case, and the width, height, and depth in the 3D network one. As shown in Table 6, a wide variety of simulation areas have been considered for the performance evaluation of different routing protocols. The simulation area and number of sensor nodes are related: a large number of nodes in a small simulation area indicates a dense network, whereas a small number of nodes in a large simulation area represents a sparse one [64]. Both are possible in a UWSN environment, depending on the applications considered in that particular environment. For example, routing protocols such as RLOR offer superior performances in dense networks because they are designed to consider the routing void problem. The number of nodes may also be varied depending on water-depth, energy consumption, and cluster forming technique. One routing protocol may not be suitable for another environment, and the number of nodes may need to be varied accordingly.

The performance evaluations for the reviewed routing protocols were conducted using different simulators, each with their own advantages and disadvantages. However, the simulator used for most of the reviewed protocols was not mentioned in the studies. Among the mentioned simulators, the SUNSET simulator provides a realistic representation of the UWSN environment; it supports various channel models and provides a detailed representation of the communication component and node energy [65]. Network Simulator version 2 (NS-2) has been used for QELAR and EDORQ with an aquatic environment simulation package called Aqua-Sim [66].

The number of sink nodes can be one or more in UWSN applications. Both single and multiple sinks have been considered for performance evaluations of the reviewed UWSN routing protocols. For the multiple-sink scenario, the destination can vary; meanwhile, for a single sink, the destination remains the same. Therefore, RL-based routing with multiple sinks may become more difficult than with a single sink, because the terminal state can vary.

The data packet size has impact on the performance of multihop communication in UWSNs [67]. In the reviewed routing protocols, the data packet sizes varied from 50 bytes to 1 MB. The simulation time represents another important factor for an RL-based routing protocol, because the learning of the agent improves over time. The protocol may not be effective if the simulation time terminates before the agent converges. For the performance evaluations of the reviewed protocols, simulation time was mentioned in several cases, such as QDAR, RCAR, DMARL, EDORQ, and RLOR. The simulation time represents the total time required for one round of data transmission. However, HyDRO was evaluated over a 6 day simulation, and QGDR was simulated 100 times.

In addition to simulation for performance evaluation, the efficiencies of the schemes were validated by comparing them against other well-defined and widely accepted protocols. All the reviewed routing protocols were likewise compared with other existing protocols. Since QELAR is an early RL-based UWSN routing protocols, it has been used for comparison with QDAR, HyDRO, CARMA, RCAR, and EDORQ.

VI. CHALLENGES AND OPEN RESEARCH ISSUES

The challenges and open research issues for RL-based UWSN routing protocols are highlighted in this section. Although the proposed protocols have shown significant performance improvements in routing, they can be further improved. Many challenges remain to be solved before these routing protocols can be implemented in real UWSN environments. These challenges, as well as future research issues, are discussed here.

A. NODE MOBILITY

In UWSNs, underwater nodes can be static (i.e., anchored to the ocean floor) or dynamic (i.e., floating with changing mobility). Most RL-based routing protocols consider only a scenario comprising static nodes, neglecting the node mobility. In a real underwater environment, node mobility arises because of water pressure and water current [28]. This node mobility changes the topology structure of the UWSN [68]. Moreover, AUVs have been used to collect data from underwater sensor nodes in different research works [69]–[71]. In these cases, the AUVs can be considered as the sink nodes with mobility. This scenario has not been considered in any of the reviewed schemes.

RL can be used to adapt to the mobility of both the sensor and sink nodes. Because RL algorithms can explore and learn in a network without knowing its full architecture, this feature may be suitable for the scenario with node mobility in UWSNs. The Q-table of each node can be updated with the changed location information of the neighboring nodes, along with the sink nodes at every timestamp. Therefore, if topology change occurs within the interval of two packet forwarding, nodes can update the neighboring information.

B. CONVERGENCE OF RL

In the RL algorithm, an agent must experience all possible states and actions to obtain the optimal result. Therefore, the agent must traverse the entire Q-table. When the number of states and actions increases, the Q table becomes large; thus, the agent requires a longer time to converge. In some cases, it may not converge; that is, it may become stuck in a local optima, without reaching the global optima.

When designing routing protocols, the convergence time must be considered. The number of states and actions should be minimized, to obtain an optimal result faster. Integrating other techniques such as fuzzy logic [72] can help limit the number of states. If the states are continuous values, it needs to be converted into discrete values. Otherwise, the number of states may become infinite, and the algorithm may not converge.

C. Q-TABLE INITIALIZATION

Q-table initialization significantly determines the learning speed of the RL algorithm. In most cases, Q-values are initialized to zero and updated only when the corresponding state–action pair is visited in the network. This process may slow the algorithm convergence. To speed up the convergence of the RL algorithm, Q-table initialization can be performed using several learned values from the same environment. In addition, to update the Q-table in the case of a large number of states and actions, virtual Q-value updating can be performed. This accelerates convergence, because the agent need not visit all states and actions.

D. Q-VALUE UPDATING

At the outset of the Q-learning algorithm, the agent takes actions and learns with no prior knowledge of the environment. This is one of the major drawbacks of RL and is computationally expensive in certain cases. Taking the wrong action may lead to a drastic changes in the environment.

Initializing Q-values by incorporating prior knowledge can improve the Q-learning algorithm [59] and reduce the convergence time. However, no precise rules are available for appropriately choosing the Q-value. Different researchers have adopted different techniques for Q-table initialization. Updating the Q-values by applying certain tricks can also lead to faster convergence. One such trick is to update two Q-values: one for an action and another for the corresponding opposite action [73]. This helps to increase the learning speed of the agent.

E. APPLYING A VARIETY OF RL ALGORITHMS

Although RL has been used to design various routing protocols for UWSNs, it can be seen from the reviewed protocols that only Q-learning is utilized for that purpose. RL includes a variety of algorithms, each having its own advantages and disadvantages. Q-learning is the most popular RL algorithm for solving routing problems in WSNs; however, other RL algorithms such as SARSA [74], actor-critic learning [75],

deep Q-learning [76] and more can be applied. It may happen that other RL algorithms achieve superior results to Q-learning, because they have been previously applied for routing in other wireless network scenarios.

F. SECURITY

Security issues are of major concern not only in UWSNs but in any type of wireless network. When collecting sensitive data, security problems represent a consistent threat to the UWSN. None of the afore-discussed routing protocols considered secure data transmission. Designing trust-based routing protocols is necessary, because UWSNs are widely used for military purposes, where data confidentiality is essential. Malicious attacks, unauthorized access, and data leakages should be considered when designing routing protocols, to make them applicable and effective for real-world UWSN tasks.

G. NODE DENSITY

Several routing protocols reviewed in this paper used sparse UWSNs, whilst others used dense UWSNs. In many cases, it has been observed that the performance of the algorithm deteriorates under an increasing number of nodes. In a real-world UWSN scenario, the node density can be high. Therefore, such routing protocols are not effective. This issue requires further research, to ensure that the performance of routing protocols is robust against node density variation.

VII. CONCLUSION

Routing for UWSNs is one of the most crucial issues in underwater applications. In RL, the efficiency of a system is increased with experience and time. This capability of RL algorithms has been widely considered in different wireless networking scenarios. RL has also been shown to significantly improve the design of routing protocols or UWSNs. In this article, we present an extensive survey of RL-based underwater routing protocols. The methods are discussed, and their advantages, disadvantages, and suitable application environments are presented. The reviewed protocols are further compared in terms of their key ideas, RL mechanisms, optimization parameters, and evaluation techniques. The applications of RL are also separately compared for all protocols. For future researchers, the research gaps and areas requiring critical improvement are emphasized as open research issues. The analysis, discussion, comparison, and future research directions highlighted in this investigation will provide UWSN researchers with an in-depth overview of existing routing protocols.

REFERENCES

[1] M. Jouhari, K. Ibrahimi, H. Tembine, and J. Ben-Othman, "Underwater wireless sensor networks: A survey on enabling technologies, localization protocols, and internet of underwater things," *IEEE Access*, vol. 7, pp. 96879–96899, 2019.

[2] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 257–279, Mar. 2005.

[3] H. Kaushal and G. Kaddoum, "Underwater optical wireless communication," *IEEE Access*, vol. 4, pp. 1518–1547, 2016.

[4] A. Palmeiro, M. Martin, I. Crowther, and M. Rhodes, "Underwater radio frequency communications," in *Proc. IEEE OCEANS*, Jun. 2011, pp. 1–8.

[5] C. Gabriel, M.-A. Khalighi, S. Bourennane, P. Leon, and V. Rigaud, "Channel modeling for underwater optical communication," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, Dec. 2011, pp. 833–837.

[6] P. Lacovara, "High-bandwidth underwater communications," *Mar. Technol. Soc. J.*, vol. 42, no. 1, pp. 93–102, 2008.

[7] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 72–83, Jan. 2000.

[8] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 84–89, Jan. 2009.

[9] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Marine Technol. Soc. J.*, vol. 42, no. 1, pp. 103–116, 2008.

[10] Z. Zeng, S. Fu, H. Zhang, Y. Dong, and J. Cheng, "A survey of underwater optical wireless communications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 204–238, 1st Quart., 2017.

[11] G. Schirripa Spagnolo, L. Cozzella, and F. Leccese, "Underwater optical wireless communications: Overview," *Sensors*, vol. 20, no. 8, p. 2261, Apr. 2020.

[12] C. Gabriel, M.-A. Khalighi, S. Bourennane, P. Leon, and V. Rigaud, "Channel modeling for underwater optical communication," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, Dec. 2011, pp. 833–837.

[13] S. Arnon, "Underwater optical wireless communication network," *Opt. Eng.*, vol. 49, no. 1, 2010, Art. no. 015001.

[14] I. F. Akyildiz, P. Wang, and Z. Sun, "Realizing underwater communication through magnetic induction," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 42–48, Nov. 2015.

[15] M. C. Domingo, "Magnetic induction for underwater wireless communication networks," *IEEE Trans. Antennas Propag.*, vol. 60, no. 6, pp. 2929–2939, Jun. 2012.

[16] H. Guo, Z. Sun, and P. Wang, "Multiple frequency band channel modeling and analysis for magnetic induction communication in practical underwater environments," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 6619–6632, Aug. 2017.

[17] B. Gulbahar and O. B. Akan, "A communication theoretical modeling and analysis of underwater magneto-inductive wireless channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3326–3334, Sep. 2012.

[18] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[19] D. P. Kumar, A. Tarachand, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Inf. Fusion*, vol. 49, pp. 1–25, Sep. 2019.

[20] M. Di and E. Meng Joo, "A survey of machine learning in wireless sensor netoworks from networking and application perspectives," in *Proc. 6th Int. Conf. Inf., Commun. Signal Process.*, 2007, pp. 1–5.

[21] G. Han, J. Jiang, N. Bao, L. Wan, and M. Guizani, "Routing protocols for underwater wireless sensor networks," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 72–78, Nov. 2015.

[22] T. Islam and Y. K. Lee, "A comprehensive survey of recent routing protocols for underwater acoustic sensor networks," *Sensors*, vol. 19, no. 19, p. 4256, 2019.

[23] M. Khalid, Z. Ullah, N. Ahmad, M. Arshad, B. Jan., Y. Cao, and A. Adnan, "A survey of routing issues and associated protocols in underwater wireless sensor networks," *J. Sensors*, vol. 2017, pp. 1–17, May 2017.

[24] M. Ayaz, I. Baig, A. Abdullah, and I. Faye, "A survey on routing techniques in underwater wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 34, no. 6, pp. 1908–1927, 2011.

[25] H.-H. Cho, C.-Y. Chen, T. K. Shih, and H.-C. Chao, "Survey on underwater delay/disruption tolerant wireless sensor network routing," *IET Wireless Sensor Syst.*, vol. 4, no. 3, pp. 112–121, Sep. 2014.

[26] N. Li, J.-F. Martínez, J. M. M. Chaus, and M. Eckert, "A survey on underwater acoustic sensor network routing protocols," *Sensors*, vol. 16, no. 3, p. 414, Mar. 2016.

[27] N. Z. Zenia, M. Aseeri, M. R. Ahmed, Z. I. Chowdhury, and M. S. Kaiser, "Energy-efficiency and reliability in MAC and routing protocols for underwater wireless sensor network: A survey," *J. Netw. Comput. Appl.*, vol. 71, pp. 72–85, Aug. 2016.

[28] M. Ahmed, M. Salleh, and M. I. Channa, "Routing protocols based on node mobility for underwater wireless sensor network (UWSN): A survey," *J. Netw. Comput. Appl.*, vol. 78, pp. 242–252, Jan. 2017.

[29] A. Datta and M. Dasgupta, "Underwater wireless sensor networks: A comprehensive survey of routing protocols," in *Proc. Conf. Inf. Commun. Technol. (CICT)*, Oct. 2018, pp. 1–6.
[30] S. Fattah, A. Gani, I. Ahmedy, M. Y. I. Idris, and I. A. T. Hashem, "A survey on underwater wireless sensor networks: Requirements, taxonomy, recent advances, and open research challenges," *Sensors*, vol. 20, no. 18, p. 5393, 2020.
[31] S. Khisa and S. Moh, "Survey on recent advancements in energy-efficient routing protocols for underwater wireless sensor networks," *IEEE Access*, vol. 9, pp. 55045–55062, 2021.
[32] Z. Mammeri, "Reinforcement learning based routing in networks: Review and classification of approaches," *IEEE Access*, vol. 7, pp. 55916–55950, 2019.
[33] S. Chettibi and S. Chikhi, "A survey of reinforcement learning based routing protocols for mobile ad-hoc networks," in *Recent Trends in Wireless and Mobile Networks*. Berlin, Germany: Springer, 2011, pp. 1–13.
[34] R. A. Nazib and S. Moh, "Reinforcement learning-based routing protocols for vehicular ad hoc networks: A comparative survey," *IEEE Access*, vol. 9, pp. 27552–27587, 2021.
[35] R. N. Raj, A. Nayak, and M. S. Kumar, "A survey and performance evaluation of reinforcement learning based spectrum aware routing in cognitive radio ad hoc networks," *Int. J. Wireless Inf. Netw.*, vol. 27, no. 1, pp. 144–163, Mar. 2020.
[36] M. Van Otterlo and M. Wiering, "Reinforcement learning and Markov decision processes," in *Reinforcement Learning*. Berlin, Germany: Springer, 2012, pp. 3–42.
[37] A. S. Polydoros and L. Nalpantidis, "Survey of model-based reinforcement learning: Applications on robotics," *J. Intell. Robot. Syst. Theory Appl.*, vol. 86, no. 2, pp. 153–173, May 2017.
[38] T. Degris, P. M. Pilarski, and R. S. Sutton, "Model-free reinforcement learning with continuous action in practice," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2012, pp. 2177–2182.
[39] R. Huang and G. V. Záruba, "Monte Carlo localization of wireless sensor networks with a single mobile beacon," *Wireless Netw.*, vol. 15, no. 8, pp. 978–990, 2009.
[40] S. Chettibi and S. Chikhi, "Adaptive maximum-lifetime routing in mobile ad-hoc networks using temporal difference reinforcement learning," *Evolving Syst.*, vol. 5, no. 2, pp. 89–108, Jun. 2014.
[41] W. D. Smart and L. P. Kaelbling, "Practical reinforcement learning in continuous spaces," in *Proc. ICML*, 2000, pp. 903–910.
[42] R. J. Williams and L. C. Baird, "Tight performance bounds on greedy policies based on imperfect value functions," College Comput. Sci., Northeastern Univ., Boston, MA, USA, Tech. Rep. NU-CCS-93-14, Nov. 1993.
[43] A. dos Santos Mignon and R. L. D. A. da Rocha, "An adaptive implementation of ε-greedy in reinforcement learning," *Proc. Comput. Sci.*, vol. 109, pp. 1146–1151, Jan. 2017.
[44] A. Garivier and E. Moulines, "On upper-confidence bound policies for switching bandit problems," in *Proc. Int. Conf. Algorithmic Learn. Theory*. Berlin, Germany: Springer, 2011, pp. 174–188.
[45] K. Verbeeck, A. Nowé, J. Parent, and K. Tuyls, "Exploring selfish reinforcement learning in repeated games with stochastic rewards," *Auton. Agents Multi-Agent Syst.*, vol. 14, no. 3, pp. 239–269, Apr. 2007.
[46] K. Asadi and M. L. Littman, "An alternative softmax operator for reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 243–252.
[47] T. Hu and Y. Fei, "QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 6, pp. 796–809, Jun. 2010.
[48] T. Hu and Y. Fei, "MURAO: A multi-level routing protocol for acoustic-optical hybrid underwater wireless sensor networks," in *Proc. 9th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. (SECON)*, Jun. 2012, pp. 218–226.
[49] Z. Jin, Y. Ma, Y. Su, S. Li, and X. Fu, "A Q-learning-based delay-aware routing algorithm to extend the lifetime of underwater sensor networks," *Sensors*, vol. 17, no. 7, p. 1660, Jul. 2017.
[50] S. Basagni, V. Di Valerio, P. Gjanci, and C. Petrioli, "Harnessing HyDRO: Harvesting-aware data ROuting for underwater wireless sensor networks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, Jun. 2018, pp. 271–279.
[51] S. Kim, "A better-performing Q-learning game-theoretic distributed routing for underwater wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 1, pp. 1–15, 2018.
[52] O. A. Karim, N. Javaid, A. Sher, Z. Wadud, and S. Ahmed, "QL-EEBDG: QLearning based energy balanced routing in underwater sensor networks," *EAI Endorsed Trans. Energy Web*, vol. 5, no. 17, Apr. 2018, Art. no. 154459.
[53] Y. Zhou, T. Cao, and W. Xiang, "QLFR: A Q-learning-based localization-free routing protocol for underwater sensor networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.
[54] V. Di Valerio, F. L. Presti, C. Petrioli, L. Picari, D. Spaccini, and S. Basagni, "CARMA: Channel-aware reinforcement learning-based multi-path adaptive routing for underwater wireless sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2634–2647, Nov. 2019.
[55] Z. Jin, Q. Zhao, and Y. Su, "RCAR: A reinforcement-learning-based routing protocol for congestion-avoided underwater acoustic sensor networks," *IEEE Sensor J.*, vol. 19, no. 22, pp. 10881–10891, Nov. 2019.
[56] S. Wang and Y. Shin, "Efficient routing protocol based on reinforcement learning for magnetic induction underwater sensor networks," *IEEE Access*, vol. 7, pp. 82027–82037, 2019.
[57] 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 Things J.*, vol. 7, no. 10, pp. 9805–9818, Oct. 2020.
[58] Y. Lu, R. He, X. Chen, B. Lin, and C. Yu, "Energy-efficient depth-based opportunistic routing with Q-learning for underwater wireless sensor networks," *Sensors*, vol. 20, no. 4, p. 1025, Feb. 2020.
[59] M. Pouyan, A. Mousavi, S. Golzari, and A. Hatam, "Improving the performance of Q-learning using simultanouse Q-values updating," in *Proc. Int. Congr. Technol., Commun. Knowl. (ICTCK)*, Nov. 2014, pp. 1–6.
[60] R. A. Nazib and S. Moh, "Sink-type-dependent data-gathering frameworks in wireless sensor networks: A comparative study," *Sensors*, vol. 21, no. 8, p. 2829, Apr. 2021.
[61] M. Ismail, M. Islam, I. Ahmad, F. A. Khan, A. B. Qazi, Z. H. Khan, Z. Wadud, and M. Al-Rakhami, "Reliable path selection and opportunistic routing protocol for underwater wireless sensor networks," *IEEE Access*, vol. 8, pp. 100346–100364, 2020.
[62] A. Wahid, S. Lee, and D. Kim, "A reliable and energy-efficient routing protocol for underwater wireless sensor networks," *Int. J. Commun. Syst.*, vol. 27, no. 10, pp. 2048–2062, 2014.
[63] A. Yahya, S. U. Islam, A. Akhunzada, G. Ahmed, S. Shamshirband, and J. Lloret, "Towards efficient sink mobility in underwater wireless sensor networks," *Energies*, vol. 11, no. 6, p. 1471, 2018.
[64] Y. Bayrakdar, N. Meratnia, and A. Kantarci, "A comparative view of routing protocols for underwater wireless sensor networks," in *Proc. IEEE OCEANS*, Jun. 2011, pp. 1–5.
[65] C. Petrioli, R. Petroccia, J. R. Potter, and D. Spaccini, "The SUNSET framework for simulation, emulation and at-sea testing of underwater wireless sensor networks," *Ad Hoc Netw.*, vol. 34, pp. 224–238, Nov. 2015.
[66] *The Network Simulator-ns-2*. Accessed: Jul. 28, 2021. [Online]. Available: https://www.isi.edu/nsnam/ns/
[67] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic, "Optimized packet size selection in underwater wireless sensor network communications," *IEEE J. Ocean. Eng.*, vol. 37, no. 3, pp. 321–337, Jul. 2012.
[68] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 11, no. 4, pp. 23–33, 2007.
[69] G. Han, X. Long, C. Zhu, M. Guizani, Y. Bi, and W. Zhang, "An AUV location prediction-based data collection scheme for underwater wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6037–6049, Jun. 2019.
[70] S.-W. Huang, E. Chen, and J. Guo, "Efficient seafloor classification and submarine cable route design using an autonomous underwater vehicle," *IEEE J. Ocean. Eng.*, vol. 43, no. 1, pp. 7–18, Jan. 2018.
[71] J. J. Kartha and L. Jacob, "Delay and lifetime performance of underwater wireless sensor networks with mobile element based data collection," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 5, p. 128757, 2015.
[72] S. Chettibi and S. Chikhi, "Dynamic fuzzy logic and reinforcement learning for adaptive energy efficient routing in mobile ad-hoc networks," *Appl. Soft Comput.*, vol. 38, pp. 321–328, Jan. 2016.
[73] Y. Song, Y.-B. Li, C.-H. Li, and G.-F. Zhang, "An efficient initialization approach of Q-learning for mobile robots," *Int. J. Control, Autom. Syst.*, vol. 10, no. 1, pp. 166–172, Feb. 2012.
[74] N. Aslam, K. Xia, and M. U. Hadi, "Optimal wireless charging inclusive of intellectual routing based on SARSA learning in renewable wireless sensor networks," *IEEE Sensors J.*, vol. 19, no. 18, pp. 8340–8351, Sep. 2019.

[75] T. Wang, S. Wu, Z. Wang, Y. Jiang, T. Ma, and Z. Yang, "A multi-featured actor-critic relay selection scheme for large-scale energy harvesting WSNs," *IEEE Wireless Commun. Lett.*, vol. 10, no. 1, pp. 180–184, Jan. 2021.
[76] Y. Su, R. Fan, X. Fu, and Z. Jin, "DQELR: An adaptive deep Q-network-based energy- and latency-aware routing protocol design for underwater acoustic sensor networks," *IEEE Access*, vol. 7, pp. 9091–9104, 2019.

REHENUMA TASNIM RODOSHI (Graduate Student Member, IEEE) received the B.Sc. degree in computer science and engineering from the University of Chittagong, Bangladesh, in 2018. She is currently pursuing the M.Sc. degree with the Smart Networking Laboratory, Chosun University, South Korea. Her current research interests include user association and resource management in cellular network architecture like cloud radio access network (C-RAN) and deep learning algorithms.

YUJAE SONG (Member, IEEE) received the Ph.D. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2016. He was a Visiting Scholar in communication systems with the KTH Royal Institute of Technology, Sweden, in 2015. Since 2016, he has been a Senior Researcher with the Maritime ICT Research and Development Center, Korea Institute of Ocean Science and Technology. His research interests include design, analysis, and optimization of various wireless communication systems, including 5G, maritime/underwater, and smart grid communications.

WOOYEOL CHOI (Member, IEEE) received the B.S. degree from the Department of Computer Science and Engineering, Pusan National University, Busan, South Korea, in 2008, and the M.S. and Ph.D. degrees from the School of Information and Communications, Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea, in 2010 and 2015, respectively. From 2015 to 2017, he was a Senior Research Scientist with the Korea Institute of Ocean Science and Technology (KIOST), Ansan, South Korea. From 2017 to 2018, he was a Senior Researcher with the Korea Aerospace Research Institute (KARI), Daejeon, South Korea. He is currently an Assistant Professor with the Department of Computer Engineering, Chosun University, Gwangju. His research interests include cross-layer protocol design, deep learning-based resource optimization, and experiment-driven evaluation of wireless networks.