Anypath Routing Protocol Design via Q-Learning for Underwater Sensor Networks

Yuan Zhou, *Senior Member, IEEE*, Tao Cao, and Wei Xiang, *Senior Member, IEEE*

***Abstract*—As a promising technology in the Internet of Underwater Things, underwater sensor networks (UWSNs) have drawn a widespread attention from both academia and industry. However, designing a routing protocol for UWSNs is a great challenge due to high energy consumption and large latency in the underwater environment. This article proposes a Q-learning-based localization-free anypath routing (QLFR) protocol to prolong the lifetime as well as reduce the end-to-end delay for UWSNs. Aiming at optimal routing policies, the Q-value is calculated by jointly considering the residual energy and depth information of sensor nodes throughout the routing process. More specifically, we define two reward functions (i.e., depth-related and energy-related rewards) for Q-learning with the objective of reducing latency and extending network lifetime. In addition, a new holding time mechanism for packet forwarding is designed according to the priority of forwarding candidate nodes. Furthermore, mathematical analyses are presented to analyze the performance and computational complexity of the proposed routing protocol. Extensive simulation results demonstrate the superiority performance of the proposed routing protocol in terms of the end-to-end delay and the network lifetime.**

***Index Terms*—Anypath routing protocol, holding time mechanism, Internet of Underwater Things (IoUT), Q-learning, underwater sensor networks (UWSNs).**

I. Introduction

THE Internet of Things (IoT), as a promising networking paradigm, can render convenient and efficient services for a wide range of application domains without manual intervention [1]–[4]. With an increasing interest in observing and exploring marine resources, the concept of IoT has extended to underwater environments, forming the so-called Internet of Underwater Things (IoUT) [5]–[7]. The IoUT is committed to providing interconnectivity among intelligent underwater devices to monitor vast unexplored underwater areas [8]. As critical infrastructure in the IoUT, underwater sensor networks (UWSNs) have found numerous underwater applications [9], such as offshore oil exploration and extraction, environmental observation for scientific exploration, ocean disaster prevention, mine recognition, and navigation assistance [10]–[14]. Fig. 1 illustrates the architecture of the UWSN. Due to the harsh underwater environment and high deployment costs, deploying UWSNs is much more challenging than deploying terrestrial wireless sensor networks (WSNs) [15]–[17].

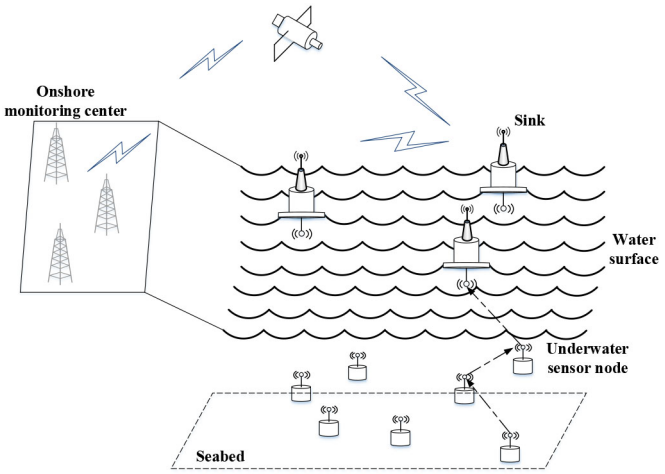


Fig. 1. Schematic of the UWSN.

Acoustic communications are preferred for UWSNs because they provide longer propagation distances [18]. However, the propagation speed of underwater acoustic waves is approximately 1500 m/s [19], resulting in large propagation latency for underwater networking services. In addition, energy efficiency has also been a major design concern for UWSNs due to the high communication energy cost [20] and the limited energy [21].

Anypath routing (also known as opportunistic routing) [22] is considered as an effective strategy for both energy efficiency and propagation latency in UWSNs. Using anypath routing, a subset of neighboring nodes is selected by the sender as the forwarding candidates according to certain criteria, e.g., energy efficiency and latency. Meanwhile, these selected forwarding candidates are assigned different priorities so that they can cooperate with each other to elect the appropriate next-hop forwarder which minimizes redundant packet transmission.

In order to address the energy efficiency problem, numerous anypath routing protocols for UWSNs favor the shortest path

Manuscript received August 21, 2020; revised October 11, 2020; accepted November 30, 2020. Date of publication December 7, 2020; date of current version May 7, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant U2006211, and in part by National Key Research and Development Project of China under Grant 2020YFC1523204. *(Corresponding author: Yuan Zhou.)*

Yuan Zhou and Tao Cao are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: zhouyuan@tju.edu.cn; caotao@tju.edu.cn).

Wei Xiang is with the School of Engineering and Mathematical Sciences, La Trobe University, Melbourne, VIC 3086, Australia, and also with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: w.xiang@latrobe.edu.au).

Digital Object Identifier 10.1109/JIOT.2020.3042901

to forward sensory data so as to minimize energy consumption [22]–[24]. However, in these methods, sensor nodes that lie in the shortest path may be overloaded with forwarding excessive packets and become hot spots. These hot spots may quickly exhaust their energy and fail prematurely, disabling the networking operations and shortening the lifetime of the entire sensor network [25]–[27]. In order to prolong the lifetime of UWSN, one should encourage data packets routing through the sensor nodes with more residual energy [25], [28], [29].

In order to tackle the latency issue, some anypath routing solutions employ greedy approaches to reduce the long delay for UWSNs [30]–[32]. However, the long-term routing rewards are not considered by the greedy approaches in these methods. Therefore, using these routing protocols, the next hop selected by the sender may not be the global optimal one for the whole routing path. In addition, these methods usually require to obtain the full-dimensional location information of sensor nodes to conduct the routing process, which may not be practical in the aquatic environment.

To simultaneously tackle the issues of low energy efficiency and high latency in UWSNs, this article proposes a novel QLFR protocol. By using the reinforcement learning (RL) (Q-learning) algorithm, the QLFR protocol takes the long-term rewards into account and thus is able to make a global optimal routing decision. Moreover, the Q-learning-based method can continuously learn from the interaction with UWSN to adapt to the complex UWSN environment. Therefore, the routing decisions become more reasonable and intelligent. Two reward functions, i.e., the depth-related and energy-related rewards, are designed for Q-learning. More elaborately, with the depth-related reward function, the proposed routing protocol does not require knowledge of the accurate full-dimensional localization of sensor nodes; instead, it only needs the depth of nodes that can be easily acquired by an inexpensive hydraulic pressure gauge. With the energy-related reward function, the nodes with more residual energy are more likely to forward data packets. As a result, the workload among sensor nodes is more balanced. At the same time, acquiring residual energy information is also easy for the nodes by checking their own battery status. Therefore, these relaxed requirements make our proposed routing protocol more practical in UWSNs. Besides, a new holding time mechanism is also devised for anypath packet forwarding. With such a mechanism, the forwarding candidate nodes are scheduled to transmit data packets in accordance with their priority levels. Furthermore, we propose a multipath suppression scheme to further reduce redundant packet transmissions and to improve energy efficiency of UWSNs.

The main contributions of this article are summarized as follows.

1) Different from other Q-learning-based routing solutions that simply select the neighboring node with the maximum Q-value as the next hop, the proposed protocol selects a set of forwarding candidates from the neighboring nodes and devises a priority mechanism according to the Q-values when making routing decisions.
2) A new holding time mechanism is designed for anypath routing, according to the priority optimized by Q-learning. To the best of our knowledge, our work is the first attempt to adopt the RL technique to design anypath routing protocols in UWSNs.
3) We propose a multipath suppression scheme to further reduce unnecessary transmissions while ensuring a high packet delivery ratio (PDR).

The remainder of this article is organized as follows. Section II provides an overview of the related work on routing protocols for UWSNs. In Section III, we first present the network topology architecture, and then model the UWSN routing problem in the general framework of Q-leaning. Section IV describes the proposed QLFR algorithm in detail and Section V elaborate on the corresponding routing protocol. A theoretical analysis of the protocol performance is presented in Section VI. In Section VII, we make a detailed complexity analysis of the proposed routing protocol. The simulation results and discussions are reported in Section VIII. Finally, concluding remarks are drawn in Section IX. Symbols used in the following sections are listed and defined in Table I.

II. Related Work

A well-designed routing strategy is very essential for reliable, fast and energy-efficient data transmission [32]–[35]. Routing protocols tailored for UWSNs have been developed for over a decade [36]–[39]. In this section, we give an overview of relevant underwater routing solutions and highlight their characteristics.

At the beginning, routing protocols in UWSNs are designed based on geographic locations of sensor nodes. Jornet *et al.* [40] proposed a distributed approach—the Focused beam routing (FBR) protocol. In this method, power control and location information are both involved in the protocol design to select the appropriate next hop forwarder. The vector-based forwarding (VBF) protocol was present in [22], in which data packets are forwarded in a virtual pipeline with a predefined radius. The virtual pipeline is specified by the routing vector from the position locations of the source node and the its destination. To improve energy efficiency, a self-adaptation algorithm is developed, which weighs the benefit of nodes within pipeline to forward packets and makes the nodes with low benefit discard the packets. The adaptive hop-by-hop VBF (AHH-VBF) [41] was a successor of VBF, in which the radius of virtual pipeline and the transmission power level are both adaptively changed hop by hop to guarantee the transmission reliability in sparse region and optimize energy efficiency. Coutinho *et al.* proposed GEDAR [37] that employs the greedy forwarding strategy to route data packets. This strategy is executed with the location of the current sender, its neighbors, and the sink node on the water surface, to determine the eligible neighboring nodes to continue forwarding the data packet toward the sink. Although these algorithms show decent performance, they assume to know the accurate 3-D location of the underwater sensor nodes. Due to the quick attenuation of radio waves in water, the GPS-based localization devices can not be applied in the UWSNs. Besides, the underwater environment is harsh and highly dynamic. Therefore, acquiring accurate 3-D location information for nodes is still a challenge in UWSNs [42], [43].

TABLE I
SYMBOL LIST

Parameters	Definition
$\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}$	Set of states, actions, transition probabilities and rewards in reinforcement learning theory, respectively
S_t, A_t	State and action in timestep t
r, π	Reward function and policy in reinforcement learning theory
$V_\pi(s), V(s)$	Expected reward when starting in state s, following policy π; and the corresponding approximated value in Q-learning
$Q_\pi(s,a), Q(s,a)$	Expected reward after taking action a in state s, following policy π; and the corresponding approximated value in Q-learning
γ, α	Discount factor and learning rate in calculating Q-value
c_e	Residual energy-related reward function
c_d	Depth-related reward function
$e_{\rm res}(s_i)$	Residual energy of node s_i
$e_{\rm ini}(s_i)$	Initial energy of node s_i
$depth(s_i)$	Depth of node s_i
$d(s_i, s_j)$	Difference between the depth of node s_i and s_j
n	The sequence number of a sensor node in the priority list
τ	Holding time
k, b	Parameters of holding time
R	Maximal communication range of a node
v_0	Speed of sound waves in aquatic environment
$t_{\max}$	Maximal propagation delay in one hop
v	Movement speed of nodes in UWSN
$A(l,f)$	Attenuation of underwater acoustic signal with frequency f kHz at transmission distance l meters
$\overline{SNR(l,f)}$	Average signal-to-noise ratio
$p(l,f,M)$	Packet delivery probability when the size of packet is M bits
$P_{s_i-\rm sink}$	Delivery probability from s_i to sink
$D_{s_i s_j}$	Distance between the node s_i and s_j
$T_{s_i-\rm sink}$	End-to-end delay from s_i to sink
λ_{s_i}	Outgoing traffic of node s_i
E_{s_i}	Energy consumption of node s_i
Γ_{s_i}	Lifetime of node s_i
$\Gamma_{\rm net}$	Network lifetime
ADD	Addition operations
MUL	Multiplication operations
L	Depth of the UWSN
ρ	Density of nodes in the whole UWSN
$\mathcal{N}_{\rm hop}$	Hop count in each packet transmission round
$\mathcal{M}_{\rm round}$	Number of packet transmission rounds

To suit the unique property of UWSNs, many localization-free routing protocols have been emerged in recent years. Yan *et al.* [44] proposed the depth-based routing (DBR) protocol, which is the first UWSN routing solution that exploits depth of sensor node to forward sensory data. Moreover, a holding time mechanism is designed to help coordinating the transmission of forwarding candidates. Wahid and Kim [45] extended DBR protocol to an energy efficient DBR (EEDBR) protocol, in which both the residual energy and the depth information of nodes are taken into consideration for selecting the next-hop forwarder. Noh *et al.* [46] reported the hydrocast anypath routing protocol which also uses depth to advance data packets. In hydrocast, the priority of next-hop node is determined based on the tradeoff between link cost and progress of the packet toward the surface. Coutinho *et al.* [47] proposed an energy balancing opportunistic routing protocol EnOR that take both packet advancement and workload balancing into consideration. A pressure-based anypath routing dubbed VAPR [30] combined depth information and hop count to set up next hop data forwarding direction, building a directional routing path to the closest sink. The next-hop forwarding set is selected according to the current data forwarding direction and next-hop data forwarding direction. Guan *et al.* [48] suggested a distance-vector-based opportunistic routing (DVOR) that uses the hop counts of sensor node toward the destination to seek the shortest routing path. Based on the hop counts, a holding time mechanism is also devised to schedule the packets forwarding. Both VAPR and DVOR used periodic beacons to dynamically establish the routing path, which will cause significant overheads to UWSNs. In [49], localization-free anypath routing and duty-cycling were symbiotic designed to achieve reliable data transmission and improve energy efficiency for UWSNs. Coutinho *et al.* [50] combined power control and localization-free anypath routing for UWSNs to simultaneously improve data delivery reliability and reduce energy consumption. Wang *et al.* [26] proposed an energy-aware and void-avoidable routing protocol (EAVARP). EAVARP use a layering phase to establish the route from the source to the sink. During the data collection phase, the sender adopts the single-path strategy and selects the neighbor with the highest priority to continue forwarding the packet.

With the dramatic advance of artificial intelligence (AI), AI-based technique has become a promising solution to design routing protocols [51]–[54]. Mao *et al.* adopted the deep learning methods to design routing algorithms for both disaster recovery networks [55] and software defined wireless networks [56]. RL has also been well exploited for designing routing protocols in WSNs [57]–[60], and more recently for routing solutions in UWSNs [61]–[63]. The *Q*-learning-based adaptive routing (QELAR) protocol proposed in [61] proved that *Q*-learning performs well in UWSNs. QELAR defined the reward function based on the residual energy of each sensor nodes and the energy distribution among neighboring nodes. In QELAR, the sender always selects the neighbor who has more residual energy as the next-hop forwarder, so as to extend the lifetime of UWSN. Moreover, to improve transmission reliability, a retransmission mechanism after transmission failures is used in QELAR. The complex underwater acoustic channel brings very high error rate for UWSN communications, which will lead to excessive retransmissions in QELAR. *Q*-learning technique was also used to select the most promising forwarders so as to reduce the long latency for UWSNs in [62] and [63]. Su *et al.* [64] reported a deep *Q*-network-based routing protocol (DQELR) for UWSNs. DQELR is a

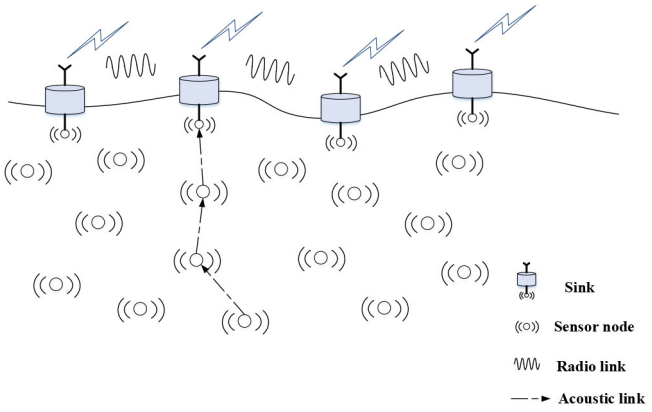


Fig. 2. Multiple-sink UWSN architecture.

single-path routing solution likewise. With the residual energy and delay takes into consideration, DQELR can prolong network lifetime and reduce the long transmission latency to some extent.

In this article, we design a localization-free anypath routing protocol with the RL technique (Q-learning) to extend the network lifetime and reduce the end-to-end delay for UWSNs. The proposed protocol selects a set of forwarding candidates from the neighboring nodes and devises a priority mechanism according to the Q-values when making routing decisions. In this case, packet retransmission occurs only if it is not received by any candidate. Therefore, the proposed method can avoid the huge energy consumption and transmission delay caused by excessive retransmissions.

III. System Model

This section first introduces the network scenario, and then models the UWSN routing problem in the general framework of Q-learning.

A. Network Scenario

In this article, we consider a multiple-sink UWSN architecture. The considered network architecture is a commonly used simulation network scenario in the UWSN and has been adopted by most of the literature proposed for UWSN routing protocol design [30], [37], [38], [44]–[49], [61]. This network architecture reasonably simplifies the practical application scenario of UWSN for the convenience of research and simulation. A schematic of such UWSN is shown in Fig. 2. The network consists of a set $N = SN \cup SK$ of sensor nodes with a maximal transmission range of R, where SN denotes the set of underwater sensor nodes, and SK represents the set composed of sink nodes.

The underwater sensor nodes $SN = \{s_1, s_2, s_3, \ldots, s_{|SN|}\}$ are randomly deployed in 3-D underwater area, equipped with the acoustic modems and sensing devices to carry out observing and exploring tasks. While the sinks $SK = \{s_{|SN|+1}, s_{|SN|+2}, s_{|SN|+3}, \ldots, s_{|SN|+|SK|}\}$ are deployed on the water surface, equipped with acoustic and radio-frequency (RF) modems. Acoustic channels are used for the underwater communication (i.e., the communication between underwater sensor nodes as well as between underwater sensor nodes and sinks), and radio channels are used for the maritime communication. Underwater sensor nodes collect data from monitoring areas and delivery the data to sinks which are considered as the destinations of underwater data packets. Afterward, sinks transmit the aggregated sensory data to satellites using radio channel, then satellites send the received data to the onshore data centers. Since the sinks can communicate to each other via the radio channels efficiently, it is reasonable to assume that a packet is delivered successfully if it arrives at any of the sink node.

B. Q-Learning Framework for UWSNs

Q-learning is a value function-based RL algorithm [65]. Unlike traditional machine learning algorithms, RL involves learning what to dol'd'–how to map states to actions, so as to maximize a numerical reward signal. Using RL method, a learning system can achieve a specific goal by utilizing its experience learning from interaction in decision-making processes.

A RL task satisfied the Markov property can be considered as a Markov decision process (MDP) [66]. A particular MDP corresponds to a tuple $<\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}>$, where $\mathcal{S}$ and $\mathcal{A}$ denote the sets of state and action, respectively; in addition, $\mathcal{P}$ and $\mathcal{R}$ represent the transition probability and reward sets, respectively, [67]. In RL, a direct reward r_{t+1} is defined as the return after taking an action at time t; a policy π is a mapping operation defined as $\pi : (s, a) \mapsto \pi(a|s)$, where $\pi(a|s)$ denotes the probability of taking action a in state s.

Then we can define two value functions, the V-value and the Q-value. Informally, the V-value for state s, denoted as $V_\pi(s)$, is the expected cumulative reward when starting in state s and following policy π thereafter. Thus, $V_\pi(s)$ can be defined as

$$V_\pi(s) \triangleq \mathbb{E}_\pi\left[\sum_{i=0}^{+\infty} \gamma^i r_{t+i+1} \big| S_t = s\right] \tag{1}$$

where $\mathbb{E}_\pi[\,\cdot\,]$ denotes the expected value following policy π, $\gamma \in [0, 1]$ is a discount factor.

Similarly, the Q-value $Q_\pi(s, a)$ for a state-action pair (s, a) denotes the expected cumulative reward of taking action a in state s by following policy π, and is defined as:

$$Q_\pi(s, a) \triangleq \mathbb{E}_\pi\left[\sum_{i=0}^{+\infty} \gamma^i r_{t+i+1} \big| S_t = s,\, A_t = a\right]. \tag{2}$$

In Q-learning, the Q-value can be approximated recursively as follows:

$$\begin{aligned} Q_{k+1}(S_t, A_t) \leftarrow\ & \alpha\big[(r_{t+1})_{k+1} + \gamma V_k(S_{t+1})\big] \\ & + (1-\alpha) Q_k(S_t, A_t) \end{aligned} \tag{3}$$

where k denotes the iteration number, $\alpha \in (0, 1]$ represents the learning rate. $V_k(S_{t+1}) = \max_{a\in\mathcal{A}} Q_k(S_{t+1}, a)$ is the approximated V-value of a given state S_{t+1}. For the sake of understanding (3), let us consider an example. Given that following policy π, we have performed a RL task k times and gotten the corresponding experiences, namely the V-value of each state and the Q-value of each state-action pair. Then

in the $(k+1)$th iteration, the Q-value in each timestep can approximated by (4) as follows:

$$\begin{aligned} Q_{k+1}(S_t, A_t) &= \alpha\left[(r_{t+1})_{k+1} + \gamma V_k(S_{t+1})\right] + (1-\alpha)Q_k(S_t, A_t) \\ &= \alpha\left[(r_{t+1})_{k+1} + \gamma \max_{a\in\mathcal{A}} Q_k(S_{t+1}, a)\right] \\ &\quad + (1-\alpha)Q_k(S_t, A_t). \end{aligned} \tag{4}$$

For the routing problems in UWSNs, we regard the entire network as a learning system. The system state can be defined as s_i when the node s_i is going to send a packet. We define a_j as the action that a packet is sent to node s_j. We consider the routing path as the policy π, because a selected routing path can direct packet forwarding a which is regarded as an action.

IV. QLFR Algorithm

This section first gives an overview of the proposed QLFR algorithm and then describes it in detail, including the reward functions, a new holding time mechanism and a multipath suppression scheme.

A. QLFR Overview

In the proposed QLFR, a routing decision should be made by the sender before it sends a packet. To select the optimal next-hop forwarder, QLFR adopts the Q-learning algorithm to calculate the Q-values of all the sender neighbors. The larger Q-value a node has, the more suitable will be for forwarding the packet.

After computing the Q-values, the sender will select the neighboring nodes whose depth is smaller than its, and sort these selected nodes in accordance with their Q-values. Then a priority list is created by the sender, in which the selected nodes are sorted in descending order of their Q-values. Afterward, this list is embedded in the sending packet.

When a node receives the packet, it verifies if its ID is in the list. If so, the node will hold the packet for a period of time dubbed the holding time. Otherwise, it discards the packet. A node with a higher priority will hold the packet for a shorter time and thus send the packet earlier than its peers. For the others, if they overhear the packet transmitted by a higher priority node during their holding time, they will abandon this packet.

B. Reward Function

In QLFR, we assign the priority for forwarding candidate nodes according to the Q-value during the data packet transmissions. As we discussed in Section III, Q-value is defined as the expected cumulative reward, so how to define the direct reward is crucial to our proposed QLFR algorithm.

In order to reduce latency and prolong the network lifetime, both the depth and residual energy information of nodes are taken into account to devise the reward functions. Suppose that node s_i send a data packet to s_j, the reward function in this scenario can be defined as

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

where $c_e(\cdot)$ is a residual energy-related cost function and $c_d(\cdot)$ is a depth-related cost function. In our algorithm, after a sender transmit a data packet, it will receive a negative reward. Both of these two cost functions are in the range of [0, 1].

The residual energy-related reward function $c_e(s_i)$ is defined by taking the residual energy of s_i into consideration, and we formulate it as

$$c_e(s_i) = 1 - \frac{e_{\text{res}}(s_i)}{e_{\text{ini}}(s_i)} \tag{6}$$

where $e_{\text{ini}}(s_i)$ and $e_{\text{res}}(s_i)$ denote the initial and residual energy of node s_i, respectively. As observed from (6), we can know that the less residual energy s_i has, the higher $c_e(s_i)$ will be, and then results in a smaller Q-value.

The depth-related cost function $c_d(s_i, s_j)$ is defined based on the depth difference between a sender s_i and its neighboring node s_j and thus can represent the end-to-end delay. It can be formulated as

$$c_d(s_i, s_j) = \frac{1}{2}\left(1 - \frac{d(s_i, s_j)}{|d|_{\max}}\right) \tag{7}$$

where $d(s_i, s_j)$ is the depth difference between s_i and s_j, $|d|_{\max}$ represents the maximum of $|d(s_i, s_j)|$. Obviously, the larger depth s_j has, the higher $c_d(s_i, s_j)$ will be.

According to the definition of reward functions above, we can get the corresponding Q-value in different packet transmission rounds via an iterative way according to (3). A packet transmission round means that the packet is routed from the source node to any of the sink nodes. Specifically, the Q-value in $(k+1)$th round packet transmission can be calculated as (8), shown at the bottom of the page.

In QLFR, before transmitting a data packet, the sender creates a priority list for its next hop forwarding candidates according to the Q-value, and embeds the priority list into the data packet. The sequence number of a candidate node in the priority list, denoted as n, represents its priority level. For example, the first node in the priority list, namely $n = 1$, means that it has the highest priority; $n = 2$ means the node has the next highest priority, and so on. After receiving the data packet, a node first retrieve the priority list. If the node finds its own ID in the priority list, it will perform according to the proposed holding time mechanism based on its priority level, which will discuss elaborately in next part; otherwise, it will discard the packet.

$$\begin{aligned} Q_{k+1}(s_i, a_j) &= \alpha\left[\left(r_{s_i s_j}^{a_j}\right)_{k+1} + \gamma \max_{a\in\mathcal{A}} Q_k(s_j, a)\right] + (1-\alpha)Q_k(s_i, a_j) \\ &= \alpha\left[\left(r_{s_i s_j}^{a_j}\right)_{k+1} + \gamma V_k(s_j)\right] + (1-\alpha)Q_k(s_i, a_j) \end{aligned} \tag{8}$$

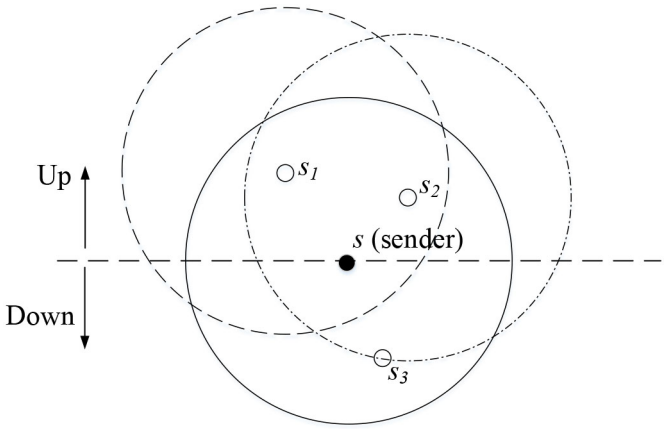


Fig. 3. Example of the holding time mechanism.

C. Holding Time

As mentioned earlier, we use holding time to schedule packet transmissions among the forwarding candidates in the proposed method. Specifically, the holding time of a node s to hold a packet is computed based on its sequence number n in the priority list. This sequence number represents the priority of s. To select a better routing and reduce redundant transmissions, the node with the higher priority should be chosen to forward the packet, meanwhile, other lower priority nodes should be prevented from forwarding the same packet.

Therefore, we can express the holding time of a node s with a linear function of n as follows:

$$\tau(n) = kn + b. \tag{9}$$

For ease of understanding, we give an simple example shown in Fig. 3 to explain the holding time mechanism. When s sends a packet, its neighbors s_1, s_2, and s_3 will receive this packet without considering the transmission failure. Then s_3 will drop the received packet since it is below s. While node s_1 and s_2 are both qualified candidates. Assuming that s_1 and s_2 receives the packet at time t_1 and t_2, respectively; the propagation delay between s_1 and s_2 is denoted as t_{prop}. In addition, supposed that $Q(s, s_1) > Q(s, s_2)$ and thus s_1 is preferred to forward the packet. Then s_2 will abandon the packet if it overhears this packet forwarded by s_1 during its holding time. Let n_1 and n_2 be the sequence numbers of nodes s_1 and s_2 in the priority list, respectively.

According to these analyses, the difference of holding times between two neighbors (e.g., s_1 and s_2) should be long enough so as to ensure that the node with lower priority can overhear the forwarding of another one before it forwards the packet on schedule. Therefore, the following two constraints need to be satisfied:

$$\tau(n_1) < \tau(n_2) \tag{10}$$

$$t_1 + \tau(n_1) + t_{\text{prop}} \le t_2 + \tau(n_2). \tag{11}$$

Substituting $\tau(n) = kn + b$ into the above inequations (10) and (11), one has

$$k \ge \frac{t_1 + t_{\text{prop}} - t_2}{n_2 - n_1}, (k > 0). \tag{12}$$

Here, k is positive. As long as the above inequation (12) holds, constraints (10) and (11) can be both satisfied. Let v_0 denote the speed of sound waves in aquatic environment and R denote the maximal communication range of a node, the maximal propagation delay in one hop can be represented by $t_{\max} = (R/v_0)$. Thus, $t_1 - t_2 \le (R/v_0)$ and $t_{\text{prop}} \le (R/v_0)$. Substituting them to (12), we have

$$\frac{t_1 + t_{\text{prop}} - t_2}{n_2 - n_1} \le \frac{2 \cdot \frac{R}{v_0}}{n_2 - n_1} = \frac{2t_{\max}}{n_2 - n_1}. \tag{13}$$

Let us set $k = (2t_{\max}/h)$, $h \in \mathbb{N}^*$. When $n_2 - n_1 \ge h$, inequation (12) will hold and one can guarantee that s_1 sends a packet earlier than s_2 and prevents s_2 from forwarding the same packet.

In the proposed routing protocol, in order to reduce the long delay, we set the holding time of the node with the highest priority to be zero. Considering that the sequence number n of such node is 1, one has

$$\begin{aligned} \tau(1) &= k \cdot 1 + b = 0 \\ b &= -k. \end{aligned} \tag{14}$$

Substituting k and b obtained above into (9), the holding time $\tau(n)$ can be represented with the following formula, and it is given by:

$$\tau(n) = k \cdot (n-1) = \frac{2t_{\max}}{h}(n-1), \ \left(h \in \mathbb{N}^*\right). \tag{15}$$

D. Multipath Suppression Scheme

In practice, we tend to choose a small k appropriately. With a smaller k, the holding times of nodes are shorter, which reduces the end-to-end delay. Meanwhile, more nodes with similar holding times will have a chance to forward the same packet, and it inevitably leads to redundant transmissions and huge energy consumption. In order to improve energy efficiency, the packet transmissions should be further restricted in the routing process.

However, if we suppress the packet transmissions too much, the PDR will be reduced. The PDR represents the ratio of the number of successfully received packets to that of the generated packets. It reflects the reliability of packet transmission. Thereby, to improve the energy efficiency while keeping high transmission reliability, we propose a new multipath suppression scheme. The procedure of the proposed scheme is shown in Algorithm 1.

We first initialize the length of the priority list as a constant integer. In addition, a threshold of the PDR is adopted to act as the balance point of energy consumption and transmission reliability. The threshold can be set according to the practical application scenarios of UWSN. In this article, we set it as 0.95 to ensure a high delivery ratio.

During packet transmission phase, the number of generated packets will be attached to the sending packet by the source. On receiving the packet, the sink can calculate the PDR by dividing the number of successfully delivered packets to the number of generated packets.

If the delivery ratio is greater than the threshold, the nodes who are qualified to forward the packet will shorten the length

Algorithm 1 Proposed Multipath Suppression Scheme. $\mathcal{M}_{\text{gen}}$ Denotes the Total Number of Generated Data Packets. $\mathcal{M}_{\text{rec}}$ Is the Number of Successfully Received Data Packets. *PDR* Represents the Current PDR. P_{th} Is a Threshold Set According to the Application Scenario

```
Initialize the length of the priority list to be a constant integer
while packet transmission phase is not completed do
    Start a new round of packet transmission
    The source attaches M_gen to the sending packet
    The sink calculates the packet delivery ratio via PDR = M_rec / M_gen
    if PDR > P_th then
        Shorten the length of priority list during the next round transmission
    else
        Increase the length of priority list during the next round transmission
    end if
end while
```

of the priority list during the next transmission round to improve the energy efficiency.

Otherwise, the sink will inform the source by broadcasting a message of lengthening the list. The source then attaches this message to the sending data packet. Thus, in the next transmission round, the eligible forwarders will increase the length of its priority list according to the message to improve the delivery ratio.

V. Routing Protocol Design

In this section, we elaborate our routing protocol from three aspects, involving packet structure, routing knowledge exchange and data packet forwarding.

A. Packet Structures

The packet structure in the network is illustrated in Fig. 4. The packet header includes three parts: 1) packet identification fields; 2) routing information fields; and 3) the information about the priority list.

The packet identification fields are as follows.

1) *Source ID*, the identifier of the source node,
2) *Packet sequence number*, the unique ID of the packet.

They are depended on the source nodes of the packet. These two fields are used to differentiate packets in data forwarding. Generally, they are permanent during the entire lifetime of the packet.

Before sending the packet, a node should embed its information in these fields, which include the following.

1) *V-Value*, the V-value of the current node.
2) *Depth*, the depth information of the current node.
3) *Residual energy*, the residual energy information of the current node.
4) *Sender ID*, the ID or address of the current node.

Upon receiving a data packet, every node retrieves these fields from packet header and updates its neighbors' information with the latest routing information which helps them making optimal routing decisions.

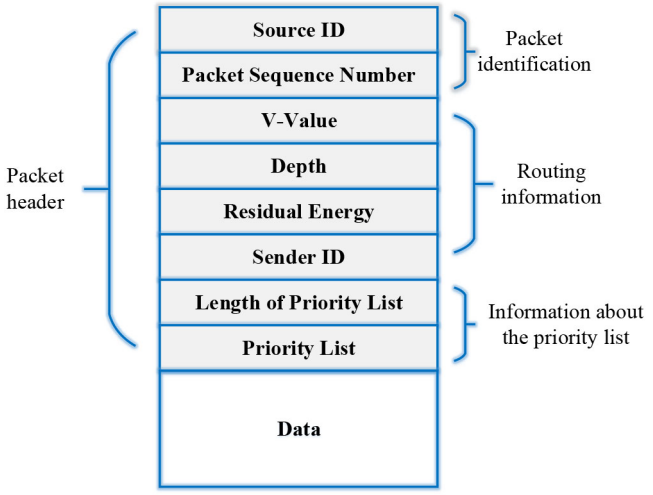


Fig. 4. Packet structure.

The information about the priority list is used to advance packet and to help the forwarding candidates cooperating with each other, which are as follows.

1) *Length of priority list*, the number of nodes eligible to forward data packets.
2) *Priority list*, routing decision made by the sender.

Length of Priority is used to control the number of forwarding candidates as mentioned above. After making a routing decision, the sender puts the priority list of the forwarding candidate nodes to the field of Priority List. According to the priority list, the next-hop nodes who received the packet will decide to perform the forwarding or to drop the packet.

Other than the packet header, the *Data* is optional. This part is the message that should be sent to the destination. If *Data* is absent, the packet is used only for routing information exchanging, which will be described in the next part.

B. Routing Knowledge Exchange

To make an optimal routing decision, all the nodes are required to know their neighbors' residual energy $e_{\text{res}}(s_i)$, depth depth(s_i), and V-values $V(s_i)$ to calculate the Q-values. A tuple expressed as $<e_{\text{res}}(s_i), \text{depth}(s_i), V(s_i)>$ is used to represent the routing knowledge of a node. In the proposed routing protocol, there are two methods to exchange the routing knowledge of sensor nodes.

1) *Exchanging Simultaneously With Data Packet Transmissions:* In the proposed protocol, the routing knowledge of the sender will be attached to the header of the sending data packet before the packet transmitted. Thus, a node can obtain the routing knowledge of its neighbors from the incoming data packets.
2) *Using the Hello Packet That Contains Only the Routing Knowledge:* Every node in the UWSN will periodically broadcast a Hello packet which has no payload and is only adopted for the routing knowledge exchange. This

kind of broadcasts can be regarded a complementary approach to exchange the routing knowledge. Since each node can obtain the routing knowledge with data packet transmissions, we do not need broadcast special control packets. Thereby, we can set the broadcast period of Hello packets to be very long, and thus, the overhead of this part can be ignored.

C. Data Packet Forwarding

Based on the crucial components of the proposed routing protocol defined above, the procedure of data packet forwarding in QLFR is discussed in this part. This procedure is summarized in Algorithm 2.

When a sender prepares to transmit a data packet, it first calculates the Q-values associated with each of its neighboring nodes using the acquired routing knowledge. A priority list is formed by opting the neighboring nodes simultaneously having two characteristics: 1) the nodes are with smaller depth than the sender and 2) with large Q-values. The neighboring nodes in the priority list are regarded as the candidates for the next hop. In addition, the *Length of Priority List* in the packet header is adopted to control the number of forwarding candidates in the priority list. Before sending the packet, the sender updates the packet header with its own information and the priority list of the next hop candidates.

Upon receiving a packet, no matter whether or not a node is designated as the qualified forwarder, it extracts the routing knowledge of the sender from the packet header, and updates the corresponding neighbor information. If the node is not eligible to forward the packet, it simply drops the packet; otherwise, the node will check whether it has forwarded the data packet before.

In this case, if the packet has been forwarded by the node before, it is discarded by the node, hence, other candidate nodes will have the opportunity to forward this packet. Otherwise, the node calculates the holding time for the packet based on its sequence number in the priority list. The first node in the priority list means that it has the highest priority, and it forwards the packet immediately without waiting; while other nodes in the priority list hold the packet for the calculated holding time. During the holding time, if a node overhearing the same packet, it will give up the forwarding of this packet, as another node with higher priority has already forwarded the packet; if not, the node will transmit the packet when holding time expires.

Moreover, the training process in our method is conducted in an online and interactive manner. As mentioned above, during each packet transmission round, a sender will calculate the Q-values for each of its neighbors using the acquired routing knowledge before sending a packet.

On one hand, the calculated Q-values are used by the sender to make its routing decision; on the other hand, these Q-values are adopted to update the sender's V-value, then the V-value is acted as the routing knowledge which will be attached to the data packets for other nodes to conduct their learning process. After a round of packet transmission, the training process goes through an iteration. According to the Q-learning theory [61], [66], Q-values will get convergence after several rounds of packet transmission, and then the routing policy will also converge to the optimal one, so as to guide a sender to choose the most appropriate next hop.

Algorithm 2 Algorithm for Data Packet Forwarding in QLFR. K Is the Data Packet. s_i Is the Node That Currently Receives the Data Packet. τ Denotes the Holding Time of s_i To Hold the Packet. $Neighbor(s_i)$ Is the Set Included All the Neighboring Nodes of s_i. s_j Is an Element in the Set $Neighbor(s_i)$

```
Onhearing K
Get the sender's information from the header of K
Get the priority list from the header of K
if s_i ∉ priority list then
    Drop K
else if s_i has forwarded K then
    Drop K
else
    Calculate τ
    Calculate Q(s_i, s_j), s_j ∈ Neighbor(s_i)
    Set the new priority list for next hop candidates
    if s_i overhears K during τ then
        Drop K
    else
        Update the header of K
        Send K when τ expires
    end if
end if
```

VI. Theoretical Analysis

In this section, we first calculate the delivery probability between two nodes, which is defined as the successful probability of a packet transmitted from one node to another in our protocol, and is crucial in deriving the other performance metrics. Then, we analyze the performance of our proposed routing protocol with respect to four performance metrics, i.e., the PDR, end-to-end delay, energy consumption, and network lifetime.

A. Delivery Probability

Urick's model [68], which is a commonly used model in the UWSN, is adopted to formulate the underwater acoustic channel in this article. In this model, the attenuation of the underwater acoustic signal with frequency f (in kHz) at transmission distance l (in meter) is given by

$$A(l, f) = A_0 l^{\kappa} a(f)^l \tag{16}$$

where A_0 is a constant attenuation factor, which models the signal attenuation caused by the propagation effects, such as scattering, refraction and multipath propagation. $\kappa \in [1, 2]$ is the spreading loss factor. $a(f)$ denotes the absorption coefficient and can be calculated by the Thorpe formula [69] as follows:

$$10\log a(f) = 2.75\times 10^{-4} f^2 + \frac{44f^2}{4100+f} + \frac{0.11f^2}{1+f^2} + 10^{-3}. \tag{17}$$

Then, for an underwater acoustic link with signal frequency f and transmission distance l, the average signal-to-noise ratio (SNR) at the receiver of this link can be shown as

$$\overline{\mathrm{SNR}(l,f)} = \frac{e_b/A(l,f)}{N_0} = \frac{e_b}{N_0 A_0 l^{\kappa} a(f)^l} \tag{18}$$

where N_0 represents the power density of the noise modeled as the additive white Gaussian noise (AWGN), and e_b is the transmit energy per bit, which are both constants. In addition, Rayleigh fading is adopted to simulate small scale fading [70]–[72], in which the probability density of the SNR can be described as follows:

$$f_{\mathrm{SNR}}(l,f,X) = \frac{1}{\overline{\mathrm{SNR}(l,f)}} e^{-\frac{X}{\overline{\mathrm{SNR}(l,f)}}}. \tag{19}$$

Thereby, we can derive the probability of data transmission errors per bit using the following formula:

$$p_e(l,f) = \int_0^{\infty} p_e(X) f_{\mathrm{SNR}}(l,f,X)\, dX \tag{20}$$

where $p_e(X)$ represents the probability of data transmission errors using any selected modulation scheme at a SNR of X. Furthermore, similar to [73]–[75], the binary phase shift keying modulation is employed in this article. Thus, as in [76] and [77], the corresponding probability of data transmission errors per bit can be shown as

$$p_e(l,f) = \frac{1}{2}\left(1 - \sqrt{\frac{\overline{\mathrm{SNR}(l,f)}}{1+\overline{\mathrm{SNR}(l,f)}}}\right). \tag{21}$$

Therefore, letting the signal frequency be f kHz, the transmission distance be l meters and the data packet size be M bits, the delivery probability can be calculated as follows:

$$p(l,f,M) = (1-p_e(l,f))^M. \tag{22}$$

B. Expected Packet Delivery Ratio

The PDR describes the probability that data packets are successfully forwarded from the source node to the sink. In order to analyze the PDR, we first discuss the delivery probability of one hop. Let us assume that the current sender is s_i, and Neighbor(s_i) is the set denoting all its neighboring nodes. In our proposed routing protocol, s_i selects a subset of Neighbor(s_i) to form the forwarding candidate set $\varphi(s_i)$ and creates a priority list for these selected candidates according to their Q-values. In the priority list $(s_1, s_2, \ldots, s_{j-1}, s_j, \ldots)$, candidates are sorted in descending order of their priority levels. That is, the front candidates have higher priorities. According to (22), when the signal frequency is f kHz, the transmission distance is l meters and the data packet size is M bits, the delivery probability between sender s_i and a candidate node s_j can be given as follows:

$$p_{s_i s_j}(l,f,M) = (1-p_e(l,f))^M. \tag{23}$$

For brevity of exposition, $p_{s_i s_j}(l,f,M)$ is simply denoted by $p_{s_i s_j}$ in the remaining of this section. Moreover, a coordination scenario [78] among these selected candidates is considered in our analysis. In this scenario, if a candidate s_j is about to forward a data packet, then the following two conditions should be satisfied.

1) The packet is successfully sent from sender s_i to the forwarding candidate node s_j.
2) Transmission errors have occurred when the packet is forwarded by candidate nodes with higher priorities than node s_j.

Therefore, the probability of the packet successfully forwarded by candidate node s_j can be calculated as

$$P_{s_i s_j} = p_{s_i s_j} \prod_{k=1}^{j-1} \left(1 - p_{s_i s_k}\right) \tag{24}$$

where $p_{s_i s_j}$ is the delivery probability between sender s_i and candidate node s_j defined in (23), and $\prod_{k=1}^{j-1}(1-p_{s_i s_k})$ describes the probability that transmission errors have occurred between the sender s_i and the forwarding candidate nodes with higher priorities than candidate s_j.

A packet is successfully delivered in one hop means that the packet is correctly delivered from sender s_i to any one of its next-hop forwarding candidates (i.e., $\forall s_j \in \varphi(s_i)$). Therefore, the delivery probability of one hop in our proposed routing protocol can be calculated as follows:

$$P_{\text{one-hop}} = \sum_{\forall s_j \in \varphi(s_i)} P_{s_i s_j}. \tag{25}$$

The packet is successfully routed from s_i to the sink node means that the packet is correctly forwarded in each hop. Thereby, we can calculate the corresponding delivery probability from s_i to the sink node in a recursive manner. More specifically, we first calculate the probability of the packet forwarded by each of s_i's candidate nodes. Then, the delivery probability from a forwarding candidate node to the sink node should be calculated in the same recursive way. This process can be formulated as

$$P_{s_i-\text{sink}} = \sum_{\forall s_j \in \varphi(s_i)} P_{s_i s_j} P_{s_j-\text{sink}} \tag{26}$$

where $P_{s_i s_j}$ denotes the probability that the packet is transmitted from s_i and forwarded by s_i's candidate node s_j in the next hop as defined in (24). $P_{s_i-\text{sink}}$ and $P_{s_j-\text{sink}}$ represent the delivery probabilities from s_i and s_j to the sink node, respectively. In addition, when s_j is a sink node, $P_{s_j-\text{sink}}$ is set to be 1. This process is iterated from the sink node to the source node, and then the expected PDR can be calculated.

C. Expected End-to-End Delay

End-to-end delay describes the duration of a packet being routed from the source node to the sink node, which is also a crucial quantitative metric for evaluating the performance of routing protocols. Similar to the analysis of the PDR, we first discuss the delay of one hop. In our proposed routing protocol, the delay in one hop consists of two parts.

1) The holding time of the sender, which is the duration that the sender should wait before transmitting the packet.

2) The latency caused by the propagation of the packet from the sender to its next-hop forwarding candidates.

We consider node s_i as the current sender. The holding time of s_i in our proposed routing protocol is calculated based on its priority level. Thus, before s_i transmits the packet, the expected holding time can be calculated as

$$\begin{aligned}\tau_{s_i}^{\text{expected}} &= \sum_{\{s_k|s_i\in\varphi(s_k)\}} \tau_{s_is_k} P_{s_ks_i} \\ &= \sum_{\{s_k|s_i\in\varphi(s_k)\}} \tau_{s_is_k} p_{s_ks_i} \prod_{m=1}^{i-1}\left(1-p_{s_ks_m}\right) \end{aligned} \tag{27}$$

where s_k is one of s_i's the neighboring nodes and acts as the sender of s_i. $\tau_{s_is_k}$ represents the holding time that s_i should wait before transmitting the data packet when the packet is sent by s_k, which is defined in (15). $P_{s_ks_i}$ denotes the probability that the packet is transmitted from s_k and forwarded by s_i as defined in (24).

To calculate the propagation time of the data packet from s_i to one of its next-hop forwarding candidates s_j, we first introduce the speed model of acoustic waves in water, which is a function of the depth (or hydraulic pressure), salinity and temperature of the water [79], [80]. This speed function can be modeled as follows:

$$\begin{aligned} v_0 = &-7.139\times10^{-13}H^3T + 2.374\times10^{-2}T^3 \\ &+ 1.675\times10^{-7}H^2 - 5.304\times10^{-2}T^2 \\ &- 1.025\times10^{-2}T(S-35) + 0.163H \\ &+ 4.591T + 1.34(S-35) + 1448.96 \end{aligned} \tag{28}$$

where H (in meter) is the underwater depth, T (in degree celsius) denotes the temperature, and S (in part per thousand) represents the salinity of the water. In general, for the sake of simplicity, this propagation speed is approximated to be a constant and set to $v_0 = 1500$ m/s [81]–[84], which is also adopted in this article. Thus, the propagation time of a packet from s_i to one of its next hop forwarding candidates s_j is

$$t_{s_is_j} = \frac{D_{s_is_j}}{v_0} \tag{29}$$

where $D_{s_is_j}$ is the distance between the sender s_i and its forwarding candidate s_j. Thereby, the latency caused by the transmission from s_i to s_j can be given by

$$\begin{aligned} T_{s_is_j} &= \tau_{s_i}^{\text{expected}} + t_{s_is_j} \\ &= \sum_{\{s_k|s_i\in\varphi(s_k)\}} \tau_{s_is_k} P_{s_ks_i} + \frac{D_{s_is_j}}{v_0}. \end{aligned} \tag{30}$$

Similar to the derivations of the PDR, the expected end-to-end delay from s_i to the sink node is also derived recursively. We first calculate the delay between s_i and each of its forwarding candidates, and then the expected end-to-end delay from the forwarding candidate to the sink node should be calculated in the same recursive manner. We express the process as follows:

$$T_{s_i-\text{sink}} = \sum_{\forall s_j\in\varphi(s_i)} \left(T_{s_is_j} + T_{s_j-\text{sink}}\right) P_{s_is_j} \tag{31}$$

where $P_{s_is_j}$ denotes the probability that the packet is transmitted from s_i and forwarded by s_i's next hop candidate node s_j as defined in (24). $T_{s_i-\text{sink}}$ and $T_{s_j-\text{sink}}$ represent the expected end-to-end delay from s_i and s_j to the sink node, respectively. Especially, if s_j is a sink node, $T_{s_j-\text{sink}}$ is equal to zero. The recursive process is performed from the sink to the source, and then the expected end-to-end delay is obtainable.

D. Expected Energy Consumption

Now, we analyze the energy consumption of node s_i. For a node in an UWSN, its energy is consumed for two reasons, i.e., transmitting and receiving data packets. To analyze the energy consumed by s_i for packet transmission, we should first know the outgoing traffic of s_i (i.e., the packets forwarded by s_i), which can be also calculated in a recursive way. First, we should obtain the outgoing traffic of s_i's sender, and then calculate the probability that the traffic is forwarded by s_i afterward

$$\begin{aligned} \lambda_{s_i} &= \sum_{\{s_k|s_i\in\varphi(s_k)\}} P_{s_ks_i}\lambda_{s_k} \\ &= \sum_{\{s_k|s_i\in\varphi(s_k)\}} p_{s_ks_i} \prod_{m=1}^{i-1}\left(1-p_{s_ks_m}\right)\lambda_{s_k} \end{aligned} \tag{32}$$

where $s_k \in \text{Neighbor}(s_i)$ is a sender of s_i. $P_{s_ks_i}$ represents the probability that the packet is transmitted from s_k and then forwarded by s_i, as defined in (24). λ_{s_i} and λ_{s_k} are the outgoing traffic of node s_i and its sender s_k, respectively. Especially, for a source node, the outgoing traffic is the packets generated by itself.

After having obtained the outgoing traffic λ_{s_i}, one can calculate the packet transmission time $\delta_{s_i}^t$ as follows:

$$\delta_{s_i}^t = \lambda_{s_i}\frac{M}{\mu} \tag{33}$$

where M (in bit) is the size of a data packet and μ (in bps) denotes the data transmission rate. Therefore, given the packet transmission power Ψ_t, the energy consumption of s_i for packet transmission (i.e., $E_{s_i}^t$) can be given as

$$E_{s_i}^t = \delta_{s_i}^t \Psi_t. \tag{34}$$

Similarly, in order to analyze the energy consumption caused by packet reception, it is necessary to calculate the duration of packet reception. Thanks to the broadcast nature of the underwater acoustic channel, node s_i is able to overhear all the data packets transmitted from its neighboring nodes, even if the packets are not for itself. Thus, the amount of time that node s_i spent to receive data packets is calculated by

$$\delta_{s_r}^r = \sum_{s_j\in\text{Neighbor}(s_i)} \lambda_{s_j}\frac{M}{\mu}. \tag{35}$$

The energy consumed by node s_i for receiving packets is calculated by the power and the amount of time it spends to receive

$$E_{s_i}^r = \delta_{s_i}^r \Psi_r \tag{36}$$

where Ψ_r is the power in receiving data packets.

Taking both packet transmission and reception into consideration, we can calculate the energy consumption of node s_i as follows:

$$\begin{aligned} E_{s_i} &= E_{s_i}^t + E_{s_i}^r \\ &= \delta_{s_i}^t \Psi_t + \delta_{s_i}^r \Psi_r \\ &= \lambda_{s_i} \frac{M}{\mu} \Psi_t + \sum_{s_j \in \text{Neighbor}(s_i)} \lambda_{s_j} \frac{M}{\mu} \Psi_r. \end{aligned} \tag{37}$$

E. Expected Network Lifetime

Setting the network running time until now to be T_{run}, the average energy consumption per second for node s_i can be shown as

$$\begin{aligned} e_{s_i} &= \frac{E_{s_i}}{T_{\text{run}}} \\ &= \frac{\lambda_{s_i} \frac{M}{\mu} \Psi_t + \sum_{s_j \in \text{Neighbor}(s_i)} \lambda_{s_j} \frac{M}{\mu} \Psi_r}{T_{\text{run}}}. \end{aligned} \tag{38}$$

Therefore, the lifetime of node s_i can be estimated as

$$\begin{aligned} \Gamma_{s_i} &= \frac{e_{\text{ini}}(s_i)}{e_{s_i}} = \frac{e_{\text{ini}}(s_i)}{\frac{E_{s_i}}{T_{\text{run}}}} \\ &= \frac{e_{\text{ini}}(s_i) T_{\text{run}}}{\lambda_{s_i} \frac{M}{\mu} \Psi_t + \sum_{s_j \in \text{Neighbor}(s_i)} \lambda_{s_j} \frac{M}{\mu} \Psi_r} \end{aligned} \tag{39}$$

where $e_{\text{ini}}(s_i)$ is the initial energy of s_i. As in [24], [61], [85], and [86], we define the network lifetime as the minimum lifetime of any sensor node. This is due to the fact that the failure of a single sensor node may interrupt network traffic and disable the entire UWSN. Thereby, the network lifetime can be estimated as

$$\begin{aligned} \Gamma_{\text{net}} &= \left(\Gamma_{s_i}\right)_{\min} = \left(\frac{e_{\text{ini}}(s_i)}{e_{s_i}}\right)_{\min} \\ &= \left(\frac{e_{\text{ini}}(s_i) T_{\text{run}}}{\lambda_{s_i} \frac{M}{\mu} \Psi_t + \sum_{s_j \in \text{Neighbor}(s_i)} \lambda_{s_j} \frac{M}{\mu} \Psi_r}\right)_{\min} \\ s_i \in SN &= \left\{s_1, s_2, s_3, \ldots, s_{|SN|}\right\}. \end{aligned} \tag{40}$$

VII. Complexity Analysis

In this section, we first present a detailed complexity analysis of our proposed routing protocol. Then we analyze the impact of this computational overhead. The complexity analysis is mainly performed by calculating the number of basic operations, i.e., multiplication operations and addition operations, which are abbreviated as MUL and ADD in the remaining of this section, respectively.

With our protocol, the hop count in a packet transmission round is a variable. In order to analyze the computational complexity of the whole process of the packet transmission phase in our protocol, we should first calculate the mathematical expectation of the hop count in each transmission round.

Lemma 1: Let R be the maximal transmission range of a node. The mathematical expectation of the depth difference between the next-hop node of the data packet and the current sender is $(3/8)R$. [It means that, in each hop, the expected distance for a packet to be advanced upward is $(3/8)R$].

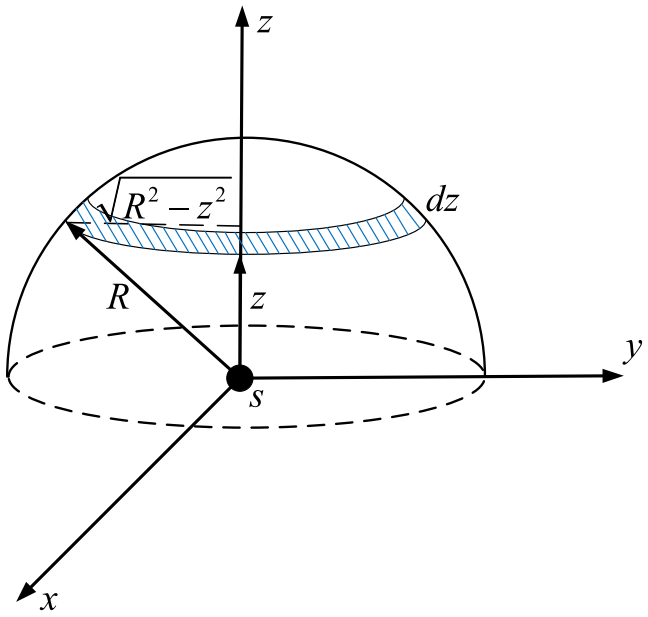


Fig. 5. Forwarding model and 3-D rectangular coordinate system with s as the origin.

Proof: Let the UWSN be a 3-D area of dimensions $L \times L \times L$. N sensor nodes are randomly deployed in the UWSN. Thus, these nodes can be considered uniformly distributed in the entire network, and the density of nodes ρ can be given by

$$\rho = \frac{N}{L^3}. \tag{41}$$

Taking the current sender s as the coordinate origin, we can establish a 3-D rectangular coordinate system as Fig. 5. In the proposed routing protocol, only the neighbors with a smaller depth than the sender have a chance to be forwarding candidates, so the packets are forwarded upward at each hop. If the coordinate of a neighboring node is (x, y, z), $(z > 0)$, then this node can advance a packet upward by z meters. The number of such neighboring nodes (i.e., the neighbors whose third coordinate are z) should be $\rho\pi(R^2 - z^2)dz$, where R is the maximal transmission range of a node. Therefore, the probability that a packet is advanced upward by z meters in one hop can be calculated as follows:

$$P(Z = z) = \frac{\rho\pi\left(R^2 - z^2\right)dz}{\int_0^R \rho\pi\left(R^2 - z^2\right)dz}, \quad 0 \le z \le R \tag{42}$$

where $\int_0^R \rho\pi(R^2 - z^2)\,dz$ denotes total number of neighboring nodes whose depth is smaller than the sender's. Hence, the corresponding cumulative distribution function is

$$F(z) = P(Z \le z) = \frac{\int_0^z \rho\pi\left(R^2 - t^2\right)dt}{\int_0^R \rho\pi\left(R^2 - z^2\right)dz}, \quad 0 \le z \le R. \tag{43}$$

The probability density function $f(z)$ is the derivative of the cumulative distribution function with respect to z, it can be calculated as

$$f(z) = F'(z) = \frac{\rho\pi\left(R^2 - z^2\right)}{\int_0^R \rho\pi\left(R^2 - z^2\right)dz}, \quad 0 \le z \le R. \tag{44}$$

Thereby, the mathematical expectation of the upward advanced distance in one hop can be derived as follows:

$$\mathbb{E}[Z] = \int_0^R zf(z)\,dz = \int_0^R z \cdot \frac{\rho\pi\left(R^2 - z^2\right)}{\int_0^R \rho\pi\left(R^2 - z^2\right)dz}\,dz$$
$$= \frac{\int_0^R z\rho\pi\left(R^2 - z^2\right)dz}{\int_0^R \rho\pi\left(R^2 - z^2\right)dz} = \frac{\rho\pi\left(\frac{R^2z^2}{2} - \frac{z^4}{4}\right)\Big|_0^R}{\rho\pi\left(R^2z - \frac{z^3}{3}\right)\Big|_0^R}$$
$$= \frac{R^4/4}{2R^3/3} = \frac{3}{8}R. \tag{45}$$

■

According to Lemma 1, a packet is expected to advance upward by $(3/8)R$ in one hop, so the mathematical expectation of the hop count in each packet transmission round can be given by

$$\mathbb{E}\left[\mathcal{N}_{\text{hop}}\right] = \frac{L}{\mathbb{E}[Z]} = \frac{L}{3R/8} = \frac{8L}{3R} \tag{46}$$

where $\mathbb{E}[\mathcal{N}_{\text{hop}}]$ represents the mathematical expectation of $\mathcal{N}_{\text{hop}}$; L is the depth of the whole UWSN; $\mathcal{N}_{\text{hop}}$ denotes the number of hops from the source node to the sink node, which is a random variable.

As shown in Algorithm 2, the computational cost of our proposed routing protocol mainly comes from two parts, i.e., calculating Q-values and holding times. We then proceed to analyze the algorithm complexity from these two aspects.

To calculate the Q-value for each neighbor, 2ADD + 3MUL is required. In each hop, the sender will calculate the Q-values associated with all its neighboring nodes. Therefore, the computational cost of calculating Q-values in one hop is

$$\int_{-R}^R \rho\pi\left(R^2 - z^2\right)dz \cdot (2\text{ADD} + 3\text{MUL})$$
$$= \frac{4}{3}\pi R^3 \frac{N}{L^3}(2\text{ADD} + 3\text{MUL}) \tag{47}$$

where $\int_{-R}^R \rho\pi(R^2 - z^2)\,dz$ denotes the number of neighboring nodes for a sender. According to (46), the expected number of hops in one transmission round is $(8L/3R)$. If there are $\mathcal{M}_{\text{round}}$ transmission rounds, the total computational complexity of calculating Q-values will be

$$\mathcal{M}_{\text{round}}\frac{8L}{3R} \cdot \frac{4}{3}\pi R^3 \frac{N}{L^3}(2\text{ADD} + 3\text{MUL})$$
$$= \mathcal{M}_{\text{round}}\frac{32\pi NR^2}{9L^2}(2\text{ADD} + 3\text{MUL}). \tag{48}$$

In our protocol, only the neighbors with a smaller depth than the sender have a chance to calculate the holding times. This process requires 1ADD+1MUL operations for each node according to (8). Thus, the computational cost of calculating holding times in each hop can be given as

$$\int_0^R \rho\pi\left(R^2 - z^2\right)dz \cdot (1\text{ADD} + 1\text{MUL})$$
$$= \frac{2}{3}\pi R^3 \frac{N}{L^3}(1\text{ADD} + 1\text{MUL}). \tag{49}$$

As a result, the total computational complexity of calculating holding times in our method is

$$\mathcal{M}_{\text{round}}\frac{8L}{3R} \cdot \frac{2}{3}\pi R^3 \frac{N}{L^3}(1\text{ADD} + 1\text{MUL})$$
$$= \mathcal{M}_{\text{round}}\frac{16\pi NR^2}{9L^2}(1\text{ADD} + 1\text{MUL}). \tag{50}$$

Although in our proposed method, each node has to carry out some computations of Q-values and holding times, these computations are simple.

Moreover, with the improvement of the computing capacity and speed of the processor, the power consumption and delay of computation are significantly reduced and are much smaller than that of acoustic communications [61]. Therefore, the computational overhead of the proposed routing protocol is ignored.

VIII. Performance Evaluation

In this, computer simulations are conducted to evaluate the performance of QLFR. The simulation configurations are first discussed. Then we compare QLFR with four other peer routing protocols to demonstrate its superiority performance. Finally, we present the parameter analysis to show how the performance of the proposed routing solution QLFR is impacted by parameters.

A. Simulation Configurations

We use MATLAB to simulate and evaluate the performance of our proposed routing solution. All the sensor nodes are randomly deployed in a 3-D area of dimensions 500 m × 500 m × 500 m in the simulations. Each node follows the random-walk mobility pattern [44]. A node will move to a new location with a given speed v after it selects a direction randomly. According to [44], [48], and [61], multiple sinks are deployed on the surface of the network, and will be stationary after they deployed. In addition, we place five source nodes at the bottom of the sensor network. The speed of sound waves in aquatic environment is $v_0 = 1500$ m/s, the maximal communication range of a node is set to $R = 150$ m; and the number of sinks is set to be five. Moreover, the values of energy consumption for the nodes' operations of packet transmission and reception are set to be $\Psi_t = 2$ W and $\Psi_r = 0.5$ W, respectively. The discount factor γ used for calculating Q-value is determined to be 0.8 as suggested in [61]. The detailed parameters used in the simulations are shown in Table II.

Besides, we adopt four quantitative metrics in our simulation experiments to examine the performance of QLFR, i.e., PDR, average end-to-end delay, network lifetime, and total energy consumption.

B. Performance Comparison

1) Benchmark Protocols: In this article, our proposed protocol QLFR is compared to four other well-known routing protocols: 1) DBR [44]; 2) EEDBR [45]; 3) DVOR [48]; and 4) QELAR [61], which represent two different paradigms of underwater routing protocols. QELAR shows how to route

TABLE II
VALUES OF PARAMETERS IN THE SIMULATIONS

Parameter	Description	Value
N	Number of sensor nodes in the entire underwater sensor network	100∼500
N_{source}	Number of source nodes in the underwater sensor network	5
N_{sink}	Number of sink nodes in the underwater sensor network	5
v_0	Speed of sound waves in aquatic environment	1500 m/s
Ψ_t	Power of packet transmission	2 W
Ψ_r	Power of packet reception	0.5 W
R	Maximal transmission range of a sensor node	150 m
γ	Discount factor of the long-term reward for calculating the Q-value	0.8
k	The difference of holding time between two adjacent nodes in the priority list	0.01 s∼0.1 s
v	Movement speed of sensor nodes in the underwater sensor network	1 m/s∼5 m/s

intelligently by using a RL technique to balance the workload of sensor nodes. DBR, EEDBR and DVOR represent the localization-free anypath routing protocols designed without the learning process. In the following, we briefly describe these four routing protocols, and highlight some of their properties.

DBR is the first UWSN routing solution that uses the depth information of sensor nodes to forward sensory data. The essential idea behind DBR is to route the packets greedily toward the destinations deployed on the water surface in terms of depth. In addition, it uses a holding mechanism to help the forwarding candidates cooperate with each other to select the closest forwarder. With the greedy strategy, DBR can reduce the average end-to-end delay of packet transmission.

EEDBR is an EEDBR protocol. In EEDBR, the next hop node is selected by first considering the residual energy of the sensor nodes. Based on residual energy, every node calculates the holding time to schedule packet forwarding. Therefore, EEDBR is an energy balanced algorithm in terms of balancing the energy consumption among sensor nodes.

DVOR is a distance-vector-based routing protocol, which uses the hop counts of sensor node toward the destination to decide on the shortest routing path. It uses a query mechanism to set up distance vectors for all nodes. The distance vectors store the least hop counts to the sink, and then data packets can be routed via the shortest path in terms of hop counts. Based on the distance vectors, DVOR can reduce detours during packet transmissions, decreasing the average end-to-end delay and energy consumption.

QELAR is a single-path routing protocol based on the *Q*-learning technique with the objective of maximizing network lifetime for UWSNs. Its reward functions take the residual energy of each sender and the energy distribution among the sender' neighboring nodes into consideration. In QELAR, the routing path are chosen for balancing the workload among sensor nodes and maximizing network lifetime. Moreover, to improve the transmission reliability, a retransmission mechanism after transmission failures is used in QELAR.

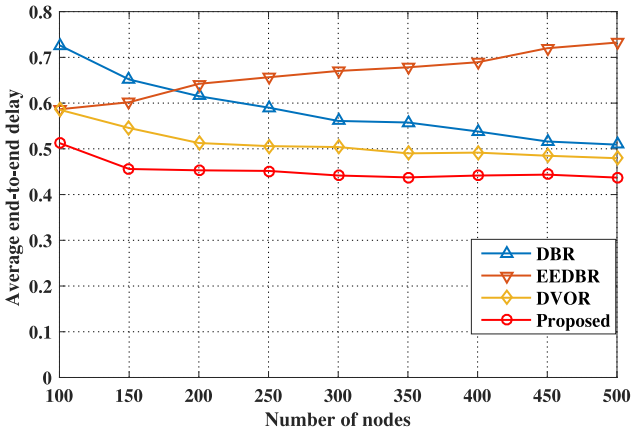


Fig. 6. Performance comparison among our proposed routing protocol, DBR, EEDBR, and DVOR in terms of average end-to-end delay.

2) Numerical Results and Discussions: First, QLFR is compared with DBR [44], EEDBR [45] and DVOR [48] with respect to three quantitative metrics, i.e., the network lifetime, average end-to-end delay, and PDR. we set k and the movement speed of a node v as 0.05 s and 3 m/s, respectively. In addition, the total number of nodes is varied from 100 to 500.

Fig. 6 shows the end-to-end delay of the four schemes. For DBR and DVOR, each node is required to hold a packet for a period of time, and thus results in the long delay in these two methods. In addition, DBR, EEDBR and DVOR use only one-hop routing information to select the next-hop candidates, which may not be appropriate from a global perspective. By contrast, for the proposed QLFR protocol, with the depth-related reward function, reducing the latency is always taken in to account during the routing process. Moreover, thanks to the proposed *Q*-learning-based algorithm, our QFLR protocol is able to make a global optimal routing decision for the whole routing path. Thereby, QLFR is superior to DBR, EEDBR and DVOR with respect to the average end-to-end delay.

The network lifetime of DBR, EEDBR, DVOR and QLFR are shown in Fig. 7. The results demonstrate that QLFR achieves the maximal lifetime among these four protocols in most situations. For DBR, lots of nodes will have the same holding time, causing excessive undesired transmissions. These undesired transmissions then lead to huge energy consumption, and thus reduce the lifetime of UWSN. DVOR favors the shortest path to route data packets, which can reduce energy consumption to some extent. However, the sensor nodes who lie in the shortest path are over-burdened and become hot spots due to transmitting too many packets. These hot spots may quickly exhaust their energy and fail prematurely, shortening the lifetime of the entire sensor network. EEDBR takes the residual energy of nodes into consideration and can prolong the network lifetime compared with DBR and DVOR.

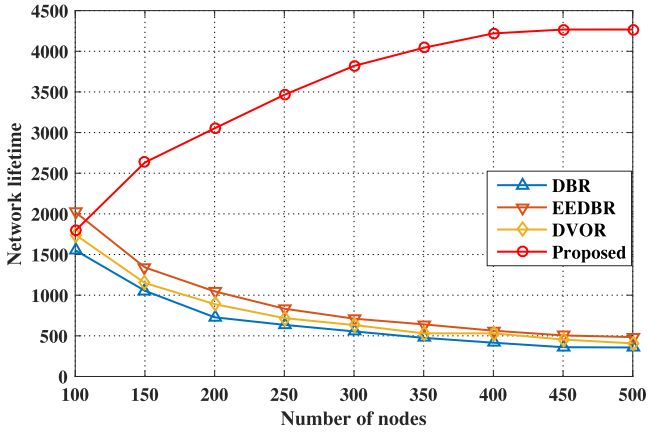


Fig. 7. Performance comparison among our proposed routing protocol, DBR, EEDBR, and DVOR in terms of network lifetime.

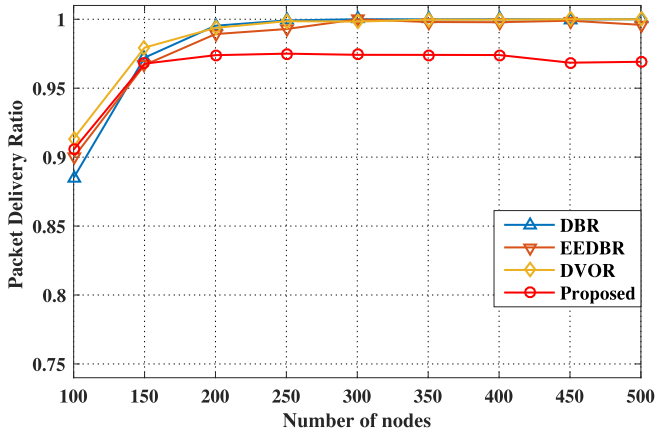


Fig. 8. Performance comparison among our proposed routing protocol, DBR, EEDBR, and DVOR in terms of PDR.

When the number of nodes is 100, the network lifetime of the proposed QLFR is slightly shorter than that of EEDBR. This is because that EEDBR aims to pursue the balance of energy distribution as much as possible. Therefore, the network lifetime can be slightly longer in EEDBR than that of other methods when the network is sparse. However, EEDBR needs to detour much to avoid nodes with lower residual energy. By contrast, with the multipath suppression scheme and the energy-related cost, QLFR can not only reduce the energy consumption, but also balance the workload among sensor nodes. Thereby, QLFR can prolong the network lifetime significantly compared with DBR, EEDBR and DVOR.

The PDR of the four methods are depicted in Fig. 8. From Fig. 8, we can observe that PDR in QLFR is just 2%–3% lower than that in DBR, EEDBR and DVOR. However, the slight improvement of the PDR in DBR, EEDBR, and DVOR sacrifices the most end-to-end delay and energy efficiency.

To conclude, the above experimental results demonstrate that the proposed routing protocol can significantly prolong the lifetime of UWSN and dramatically reduce the latency compare with DBR, EEDBR and DVOR. At the same time, QLFR can also ensure a delivery ratio similar to those three protocols.

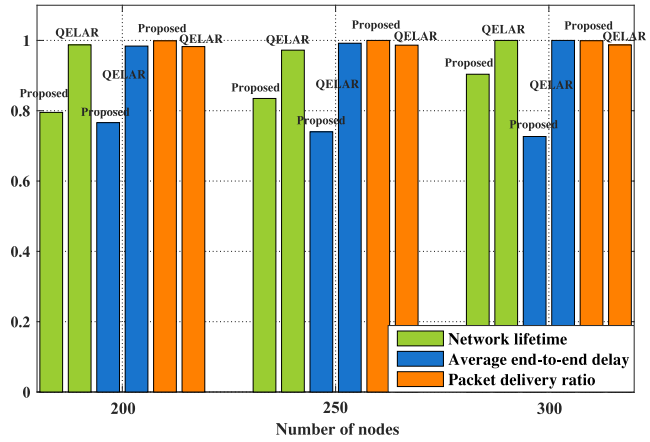


Fig. 9. Performance comparison between our proposed routing protocol and QELAR.

Now, we compare our algorithm with the Q-learning-based lifetime extended protocol, QELAR [61]. The movement speed of sensor node v is 1 m/s, and the number of nodes ranges from 200 to 300 as given in [61]. We evaluate the network lifetime, PDR, and average end-to-end delay with different node densities.

Fig. 9 shows the results concerning three metrics above. To facilitate comparison, these three metrics are all normalized within 1 according to [61]. As can be observed from the figure, our algorithm can significantly reduce latency compared to QELAR. The reason is that QELAR pursues the balance of residual energy distribution as much as possible, but does not restrict end-to-end delay. Therefore, QELAR detours in the most routing paths, which will cause a huge end-to-end delay, especially in dense networks. Furthermore, the retransmission mechanism in QELAR also increases the end-to-end delay. In contrast, by using a depth-related cost function, QLFR can reduce the latency effectively. In addition, the holding time mechanism in QLFR can avoid retransmissions, which can further improve the end-to-end delay compared with QELAR.

We can also observe from Fig. 9 that the PDR in our algorithm is 1.5% higher than that in QELAR. This is mainly due to the fact that our proposed protocol adopts the any-path routing paradigm instead of the single-path routing used in QELAR, which can achieve higher transmission reliability. Thus, the PDR in QLFR is improved than that in QELAR.

Moreover, as can be observed from Fig. 9, the network lifetime of QLFR is slightly shorter than that of QEALR. This is because QELAR can distribute workload to almost every node in UWSNs, prolonging the network lifetime greatly. While QLFR also considers reducing the transmission delay in the network. However, the network lifetime of QLFR can usually reach more than 85% of that in QELAR. Even on the worst case, it can still achieve 80.5% compared with the network lifetime of QELAR.

C. Impact of Parameters

1) Holding Time Related Parameter k: We first explore how the performance of the proposed routing protocol is impacted by the holding time related parameter k. Specifically, the

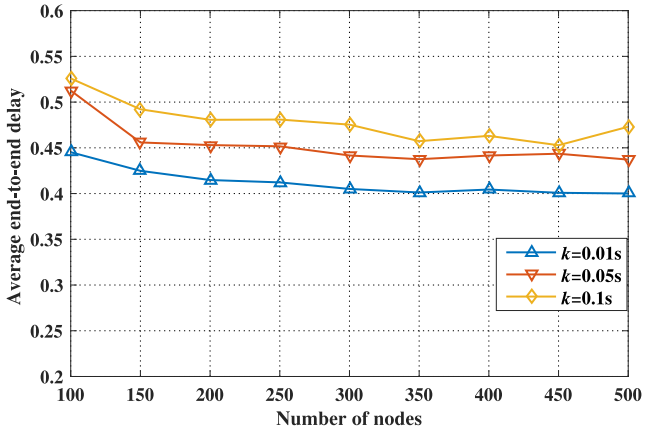


Fig. 10. Average end-to-end delay with varying k.

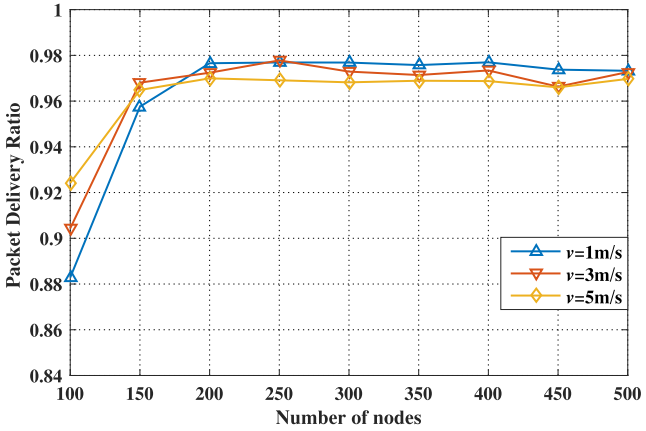


Fig. 13. PDR with varying v.

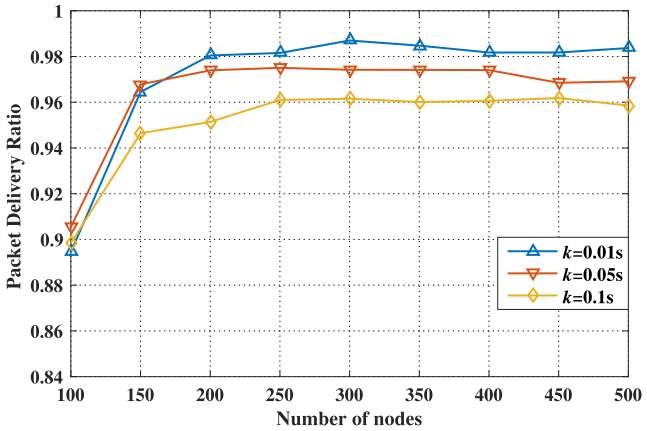


Fig. 11. PDR with varying k.

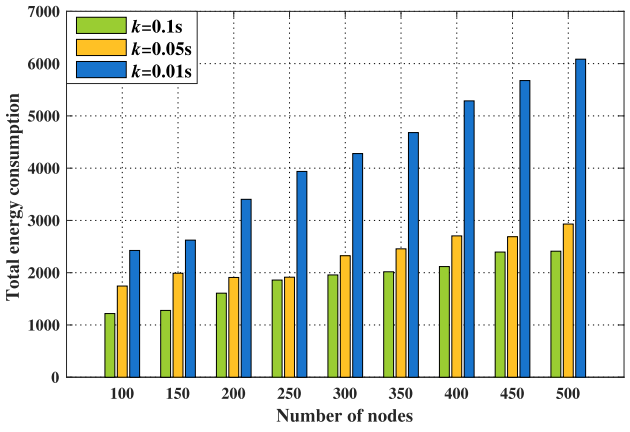


Fig. 12. Total energy consumption with varying k.

proposed QLFR is evaluated in different k values to analyze the effects of k. Setting $v_0 = 1500$ and $R = 150$ in (15), we have $k = (0.2/h)$; since $h \in \mathbb{N}^*$, we have $k \leq 0.2$. In this part of simulations, k is set to different values as 0.01 s, 0.05 s, and 0.1 s, respectively. In addition, the movement speed of sensor node v is 3 m/s, and the number of nodes ranges from 100 to 500.

Fig. 10 depicts that the average end-to-end delay positively correlates to k. According to (9) and (15), enlargeing k increases the holding time, which makes each node hold a packet for a longer time. Thus, the end-to-end delay becomes larger correspondingly.

As plotted in Fig. 11, the PDR negatively correlates to k. The reason is that decreasing k reduces the holding time of nodes, and thus makes more nodes participating in packets transmission. In this case, the PDR will be improved.

Fig. 12 reveals that the total energy consumption also negatively correlates to k. Especially, the total energy consumption of the case when $k = 0.01$ s is much more than that of others'. It is due to that, the smaller k is, the shorter holding time a node will have, which then leads to more redundant packet transmissions. Correspondingly, the energy consumption will be huger.

According to the above impact of parameter variation, we finally set k to be 0.05 s in other experimental parts for balancing the performance among PDR, end-to-end delay, and energy consumption.

2) Node Movement Speed v: As underwater sensor nodes are mobile, we examine how the movement speed v of nodes affects the performance of QLFR. To this end, we simulate QLFR with different fixed node speeds at 1 m/s, 3 m/s, and 5 m/s, respectively, [84]. At the same time, k is set to be 0.05 s, and the number of sensor nodes ranges from 100 to 500 [48].

Fig. 13 presents the PDR at different node speeds. As shown in Fig. 13, the mobility of sensor nodes can improve the PDR of the protocol, especially in sparse networks. This can be explained by the fact that when movement speeds of sensor nodes increase, the network topology will change rapidly. In this case, due to the rapid movement of nodes, the void-hole region in network can be covered rapidly and the coverage rate of the network will increase in the unit time. Therefore, if the network density is low, the delivery ratio increases with the increase of the node speed. However, if the network becomes dense, the probability of the void-hole region emerging will be reduced, hence, the movement speed of nodes has little impact on PDR.

As illustrated in Fig. 14, the average end-to-end delay is almost the same under different v. The result shows that the movement speed of nodes has little impact on the average end-to-end delay. As can be seen from Fig. 15, the total energy

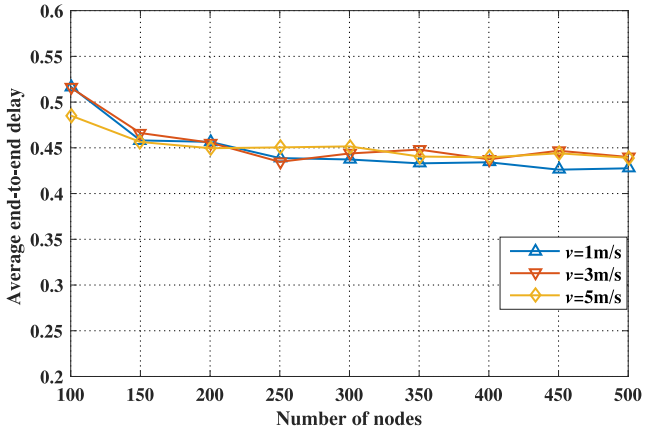


Fig. 14. Average end-to-end delay with varying v.

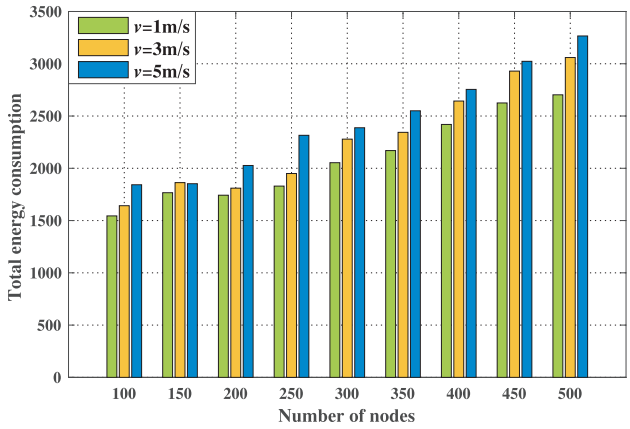


Fig. 15. Total energy consumption with varying v.

consumption only increases slightly with the movement speed of nodes. This is because the node movement can increase the coverage of the network and then slightly raise the data packet transmission rate in the network. However, the change of total energy consumption at different movement speeds of nodes is gentle and negligible.

Based on the above analysis, QLFR can well deal with the mobility of sensor nodes and is suitable for mobile UWSNs which feature highly dynamic network topology.

IX. Conclusion

This article investigated routing protocols for UWSNs and proposed a Q-learning-based localization-free anypath routing protocol dubbed QLFR. First, in order to reduce the end-to-end delay and extend the network lifetime, we designed the depth-based and energy-related rewards to calculate the Q-value, which is considered as the priority metric for forwarding candidate nodes. Then we devised a new holding time mechanism to schedule the packet transmission operations among candidates according to the priority levels of them. Finally, a multipath suppression scheme was proposed to reduce unnecessary packet transmissions and energy consumption so that the energy efficiency can be improved. Extensive simulation results were presented to demonstrate that our routing protocol outperforms the comparative routing solutions.

In this article, we focused on the two main challenges in UWSNs, i.e., energy efficiency and delay. Other concerns in UWSNs, such as bandwidth, link quality, can also be considered in the reward function for a more comprehensive routing decision, which will be our future work to explore.

References

[1] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

[2] Z. Hou, H. Chen, Y. Li, and B. Vucetic, "Incentive mechanism design for wireless energy harvesting-based Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2620–2632, Aug. 2018.

[3] T. A. Al-Janabi and H. S. Al-Raweshidy, "An energy efficient hybrid MAC protocol with dynamic sleep-based scheduling for high density IoT networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2273–2287, Apr. 2019.

[4] Z. Ma, M. Xiao, Y. Xiao, Z. Pang, H. V. Poor, and B. Vucetic, "High-reliability and low-latency wireless communication for Internet of Things: Challenges, fundamentals and enabling technologies," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7946–7970, Oct. 2019.

[5] M. C. Domingo, "An overview of the Internet of underwater things," *J. Netw. Comput. Appl.*, vol. 35, no. 6, pp. 1879–1890, 2012.

[6] M. T. R. Khan, S. H. Ahmed, and D. Kim, "AUV-aided energy-efficient clustering in the Internet of underwater things," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 4, pp. 1132–1141, Dec. 2019.

[7] M. Rahmati, V. Sadhu, and D. Pompili, "Eco-UW IoT: Eco-friendly reliable and persistent data transmission in underwater Internet of Things," in *Proc. IEEE 16th Annu. IEEE Int. Conf. Sens. Commun. Netw. (SECON)*, 2019, pp. 1–9.

[8] M. Xu and L. Liu, "Sender–receiver role-based energy-aware scheduling for Internet of Underwater Things," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 2, pp. 324–336, Apr.–Jun. 2019.

[9] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in protocol research for underwater acoustic sensor networks," in *Proc. 1st ACM Int. Workshop Underwater Netw.*, 2006, pp. 7–16.

[10] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A survey on MAC protocols for underwater wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1433–1447, 3rd Quart., 2014.

[11] C.-C. Hsu, M.-S. Kuo, C.-F. Chou, and K. C.-J. Lin, "The elimination of spatial–temporal uncertainty in underwater sensor networks," *IEEE/ACM Trans. Netw*, vol. 21, no. 4, pp. 1229–1242, Aug. 2013.

[12] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "Void-handling techniques for routing protocols in underwater sensor networks: Survey and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 800–827, 2nd Quart., 2017.

[13] H. U. Yildiz, V. C. Gungor, and B. Tavli, "Packet size optimization for lifetime maximization in underwater acoustic sensor networks," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 719–729, Feb. 2019.

[14] H. Luo, K. Wu, R. Ruby, Y. Liang, Z. Guo, and L. M. Ni, "Software-defined architectures and technologies for underwater wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2855–2888, 4th Quart., 2018.

[15] F. Liu, Y. Wang, M. Lin, K. Liu, and D. Wu, "A distributed routing algorithm for data collection in low-duty-cycle wireless sensor networks," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1420–1433, Oct. 2017.

[16] S. Han, S. Xu, W. Meng, and C. Li, "Dense-device-enabled cooperative networks for efficient and secure transmission," *IEEE Netw.*, vol. 32, no. 2, pp. 100–106, Mar./Apr. 2018.

[17] R. Monica, L. Davoli, and G. Ferrari, "A wave-based request-response protocol for latency minimization in WSNs," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7971–7979, Oct. 2019.

[18] Z. Iqbal and H.-N. Lee, "Spatially concatenated channel-network code for underwater wireless sensor networks," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3901–3914, Sep. 2016.

[19] 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.

[20] Y. Yuan, C. Liang, M. Kaneko, X. Chen, and D. Hogrefe, "Topology control for energy-efficient localization in mobile underwater sensor networks using Stackelberg game," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1487–1500, Feb. 2019.

[21] M. Ammar, K. Ibrahimi, M. Jouhari, and J. Ben-Othman, "MAC protocol-based depth adjustment and splitting mechanism for underwater sensor network (UWSN)," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–6.

[22] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-based forwarding protocol for underwater sensor networks," in *Proc. Int. Conf. Res. Netw.*, 2006, pp. 1216–1221.

[23] S. Gopi, K. Govindan, D. Chander, U. B. Desai, and S. Merchant, "E-PULRP: Energy optimized path unaware layered routing protocol for underwater sensor networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3391–3401, Nov. 2010.

[24] J. M. Jornet, M. Stojanovic, and M. Zorzi, "On joint frequency and power allocation in a cross-layer protocol for underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 35, no. 4, pp. 936–947, Oct. 2010.

[25] N. Javaid, S. Cheema, M. Akbar, N. Alrajeh, M. S. Alabed, and N. Guizani, "Balanced energy consumption based adaptive routing for IoT enabling underwater WSNs," *IEEE Access*, vol. 5, pp. 10040–10051, 2017.

[26] Z. Wang, G. Han, H. Qin, S. Zhang, and Y. Sui, "An energy-aware and void-avoidable routing protocol for underwater sensor networks," *IEEE Access*, vol. 6, pp. 7792–7801, 2018.

[27] P. Ponnavaikko, K. Yassin, S. K. Wilson, M. Stojanovic, and J. Holliday, "Energy optimization with delay constraints in underwater acoustic networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2013, pp. 551–556.

[28] K. Wang, H. Gao, X. Xu, J. Jiang, and D. Yue, "An energy-efficient reliable data transmission scheme for complex environmental monitoring in underwater acoustic sensor networks," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4051–4062, Jun. 2016.

[29] D. Sandeep and V. Kumar, "Review on clustering, coverage and connectivity in underwater wireless sensor networks: A communication techniques perspective," *IEEE Access*, vol. 5, pp. 11176–11199, 2017.

[30] Y. Noh, U. Lee, P. Wang, B. S. C. Choi, and M. Gerla, "VAPR: Void-aware pressure routing for underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 895–908, May 2013.

[31] C.-C. Hsu, H.-H. Liu, J. L. G. Gómez, and C.-F. Chou, "Delay-sensitive opportunistic routing for underwater sensor networks," *IEEE Sensors J.*, vol. 15, no. 11, pp. 6584–6591, Nov. 2015.

[32] S. Han, J. Yue, W. Meng, and X. Wu, "A localization based routing protocol for dynamic underwater sensor networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2016, pp. 1–6.

[33] H. Gao, C. Liu, Y. Li, and X. Yang, "V2VR: Reliable hybrid-network-oriented V2V data transmission and routing considering RSUS and connectivity probability," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 13, 2020, doi: 10.1109/TITS.2020.2983835.

[34] T. Baker, B. Al-Dawsari, H. Tawfik, D. Reid, and Y. Ngoko, "GreeDi: An energy efficient routing algorithm for big data on cloud," *Ad Hoc Netw.*, vol. 35, pp. 83–96, Dec. 2015.

[35] T. Baker *et al.*, "GreeAODV: An energy efficient routing protocol for vehicular ad hoc networks," in *Proc. Int. Conf. Intell. Comput.*, 2018, pp. 670–681.

[36] G. Han, S. Shen, H. Song, T. Yang, and W. Zhang, "A stratification-based data collection scheme in underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10671–10682, Nov. 2018.

[37] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro, "Geographic and opportunistic routing for underwater sensor networks," *IEEE Trans. Comput.*, vol. 65, no. 2, pp. 548–561, Feb. 2016.

[38] M. Faheem, G. Tuna, and V. C. Gungor, "QERP: Quality-of-service (QoS) aware evolutionary routing protocol for underwater wireless sensor networks," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2066–2073, Sep. 2018.

[39] R. Diamant, P. Casari, F. Campagnaro, O. Kebkal, V. Kebkal, and M. Zorzi, "Fair and throughput-optimal routing in multimodal underwater networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1738–1754, Mar. 2018.

[40] J. M. Jornet, M. Stojanovic, and M. Zorzi, "Focused beam routing protocol for underwater acoustic networks," in *Proc. 3rd ACM Int. Workshop Underwater Netw.*, 2008, pp. 75–82.

[41] H. Yu, N. Yao, and J. Liu, "An adaptive routing protocol in underwater sparse acoustic sensor networks," *Ad Hoc Netw.*, vol. 34, pp. 121–143, Nov. 2015.

[42] J. Yan, X. Zhang, X. Luo, Y. Wang, C. Chen, and X. Guan, "Asynchronous localization with mobility prediction for underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2543–2556, Mar. 2018.

[43] J. Liu, Z. Wang, J.-H. Cui, S. Zhou, and B. Yang, "A joint time synchronization and localization design for mobile underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 3, pp. 530–543, Mar. 2016.

[44] H. Yan, Z. J. Shi, and J.-H. Cui, "DBR: Depth-based routing for underwater sensor networks," in *Proc. Int. Conf. Res. Netw.*, 2008, pp. 72–86.

[45] A. Wahid and D. Kim, "An energy efficient localization-free routing protocol for underwater wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 4, 2012, Art. no. 307246.

[46] Y. Noh *et al.*, "Hydrocast: Pressure routing for underwater sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 333–347, Jan. 2016.

[47] R. W. Coutinho, A. Boukerche, L. F. Vieira, and A. A. Loureiro, "ENOR: Energy balancing routing protocol for underwater sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–6.

[48] Q. Guan, F. Ji, Y. Liu, H. Yu, and W. Chen, "Distance-vector-based opportunistic routing for underwater acoustic sensor networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3831–3839, Apr. 2019.

[49] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro, "A joint anypath routing and duty-cycling model for sustainable underwater sensor networks," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 4, pp. 314–325, Oct./Dec. 2019.

[50] R. W. Coutinho, A. Boukerche, and A. A. Loureiro, "Modeling power control and anypath routing in underwater wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2018, pp. 1–6.

[51] Q. Wang, J. Li, Q. Qi, P. Zhou, and D. O. Wu, "A game theoretic routing protocol for 3D underwater acoustic sensor networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9846–9857, Oct. 2020.

[52] B. Mao, F. Tang, Z. M. Fadlullah, and N. Kato, "An intelligent route computation approach based on real-time deep learning strategy for software defined communication systems," *IEEE Trans. Emerg. Topics Comput.*, early access, Feb. 14, 2019, doi: 10.1109/TETC.2019.2899407.

[53] B. Mao *et al.*, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[54] Z. M. Fadlullah, B. Mao, F. Tang, and N. Kato, "Value iteration architecture based deep learning for intelligent routing exploiting heterogeneous computing platforms," *IEEE Trans. Comput.*, vol. 68, no. 6, pp. 939–950, Jun. 2019.

[55] B. Mao, F. Tang, Z. M. Fadlullah, and N. Kato, "An intelligent packet forwarding approach for disaster recovery networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–6.

[56] B. Mao *et al.*, "A novel non-supervised deep-learning-based network traffic control method for software defined wireless networks," *IEEE Wireless Commun.*, vol. 25, no. 4, pp. 74–81, Aug. 2018.

[57] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 671–678.

[58] P. Wang and T. Wang, "Adaptive routing for sensor networks using reinforcement learning," in *Proc. IEEE 6th IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, 2006, p. 219.

[59] H. Kapil and C. S. R. Murthy, "A pragmatic relay placement approach in 3-D space and *Q*-learning-based transmission scheme for reliable factory automation applications," *IEEE Syst. J.*, vol. 12, no. 1, pp. 823–833, Mar. 2018.

[60] W. Jin, R. Gu, and Y. Ji, "Reward function learning for *Q*-learning-based geographic routing protocol," *IEEE Commun. Lett.*, vol. 23, no. 7, pp. 1236–1239, Jul. 2019.

[61] 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.

[62] T. Hu and Y. Fei, "MURAO: A multi-level routing protocol for acoustic-optical hybrid underwater wireless sensor networks," in *Proc. IEEE 9th Annu. IEEE Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw. (SECON)*, 2012, pp. 218–226.

[63] T. Hu and Y. Fei, "An adaptive routing protocol based on connectivity prediction for underwater disruption tolerant networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2013, pp. 65–71.

[64] 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.

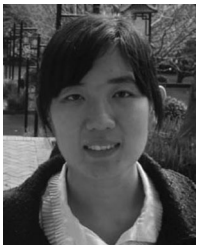
[65] C. J. Watkins and P. Dayan, "*Q*-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.

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

[67] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, pp. 237–285, May 1996.

[68] R. J. Urick, *Principles of Underwater Sound for Engineers*. New York, NY, USA: Tata McGraw-Hill Educ., 1967.

[69] W. H. Thorp, "Analytic description of the low-frequency attenuation coefficient," *J. Acoust. Soc. America*, vol. 42, no. 1, p. 270, 1967.
[70] M. Stojanovic, "Recent advances in high-speed underwater acoustic communications," *IEEE J. Ocean. Eng.*, vol. 21, no. 2, pp. 125–136, Apr. 1996.
[71] H. Yang, B. Liu, F. Ren, H. Wen, and C. Lin, "Optimization of energy efficient transmission in underwater sensor networks," in *Proc. GLOBECOM IEEE Global Telecommun. Conf.*, 2009, pp. 1–6.
[72] F. A. de Souza, R. D. Souza, G. Brante, M. E. Pellenz, and F. Rosas, "Code rate, frequency and SNR optimization for energy efficient underwater acoustic communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2015, pp. 6351–6356.
[73] M. Zorzi, P. Casari, N. Baldo, and A. F. Harris, "Energy-efficient routing schemes for underwater acoustic networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1754–1766, Dec. 2008.
[74] Y. Luo, L. Pu, Y. Zhao, and J.-H. Cui, "Harness interference for performance improvement in underwater sensor networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 258–269, Mar. 2019.
[75] F. de Souza, R. Souza, G. Brante, M. Pellenz, F. Rosas, and B. Chang, "Code rate optimization for energy efficient delay constrained underwater acoustic communications," in *Proc. IEEE OCEANS Genova*, 2015, pp. 1–4.
[76] T. S. Rappaport *et al.*, *Wireless Communications: Principles and Practice*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
[77] L. F. M. Vieira, "Performance and trade-offs of opportunistic routing in underwater networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2012, pp. 2911–2915.
[78] A. Darehshoorzadeh, E. Robson, and A. Boukerche, "Toward a comprehensive model for performance analysis of opportunistic routing in wireless mesh networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5424–5438, Jul. 2016.
[79] M. T. Isik and O. B. Akan, "A three dimensional localization algorithm for underwater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 9, pp. 4457–4463, Sep. 2009.
[80] B. Liu, H. Chen, Z. Zhong, and H. V. Poor, "Asymmetrical round trip based synchronization-free localization in large-scale underwater sensor networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3532–3542, Nov. 2010.
[81] L. Liu, Y. Liu, and N. Zhang, "A complex network approach to topology control problem in underwater acoustic sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 12, pp. 3046–3055, Dec. 2014.
[82] C.-Y. Huang, P. Ramanathan, and K. Saluja, "Routing TCP flows in underwater mesh networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 10, pp. 2022–2032, Dec. 2011.
[83] L. Liu, M. Ma, C. Liu, and Y. Shu, "Optimal relay node placement and flow allocation in underwater acoustic sensor networks," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2141–2152, May 2017.
[84] H. Yu, N. Yao, W. Tong, G. Li, Z. Gao, and G. Tan, "WDFAD-DBR: Weighting depth and forwarding area division DBR routing protocol for UASNs," *Ad Hoc Netw.*, vol. 37, no. P2, pp. 256–282, 2016.
[85] H. Yang, Y. Zhou, Y.-H. Hu, B. Wang, and S.-Y. Kung, "Cross-layer design for network lifetime maximization in underwater wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–6.
[86] J. W. Jung and M. A. Weitnauer, "On using cooperative routing for lifetime optimization of multi-hop wireless sensor networks: Analysis and guidelines," *IEEE Trans. Commun.*, vol. 61, no. 8, pp. 3413–3423, Aug. 2013.



Yuan Zhou (Senior Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in electronic engineering and communication engineering from Tianjin University, Tianjin, China, in 2006, 2008, and 2011, respectively.

Since 2011, she has been a Faculty Member with the School of Electronic Information Engineering, Tianjin University, where she is currently an Associate Professor. From 2013 to 2014, she was a Visiting Scholar with the School of Mechanical and Electrical Engineering, University of Southern Queensland, Toowoomba, QLD, Australia. From 2016 to 2017, she was a Visiting Scholar with the Department of Electrical Engineering, Princeton University, Princeton, NJ, USA. Her current research interests include wireless sensor networks and image/video communications.

Tao Cao received the B.Eng. degree from the Taiyuan University of Technology, Taiyuan, China, in 2018. He is currently pursuing the M.Eng. degree with the School of Electrical and Information Engineering, Tianjin University, Tianjin, China.

His research interest is in wireless sensor networks and reinforcement learning.

Wei Xiang (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1997 and 2000, respectively, and the Ph.D. degree in telecommunications engineering from the University of South Australia, Adelaide, SA, Australia, in 2004.

He was a Founding Chair and the Head of the Discipline of Internet of Things Engineering, James Cook University, Cairns, QLD, Australia. He is a Cisco Chair of AI and Internet of Things with La Trobe University, Melbourne, VIC, Australia. Since 2019, he has been an Adjunct Professor with the School of Electrical and Information Engineering, Tianjin University, Tianjin, China. Due to his instrumental leadership in establishing Australia's first accredited Internet of Things Engineering degree program, he was selected into Pearcy Foundation's Hall of Fame in October 2018. He is an Elected Fellow of the IET in U.K. and Engineers Australia. He has published over 250 peer-reviewed papers including three academic books and 180 journal articles. His research interests include the broad areas of communications and information theory, particularly the Internet of Things, and coding and signal processing for multimedia communications systems.

Prof. Xiang received the TNQ Innovation Award in 2016, the Pearcey Entrepreneurship Award in 2017, and the Engineers Australia Cairns Engineer in 2017. He was a co-recipient of four best paper awards at WiSATS'2019, WCSP'2015, IEEE WCNC'2011, and ICWMC'2009. He has been awarded several prestigious fellowship titles. He was named a Queensland International Fellow by the Queensland Government of Australia from 2010 to 2011, an Endeavour Research Fellow by the Commonwealth Government of Australia from 2012 to 2013, a Smart Futures Fellow by the Queensland Government of Australia from 2012 to 2015, and a JSPS Invitational Fellow jointly by the Australian Academy of Science and Japanese Society for Promotion of Science from 2014 to 2015. He is the Vice Chair of the IEEE Northern Australia Section. He was an Editor of IEEE COMMUNICATIONS LETTERS from 2015 to 2017, and is an Associate Editor for IEEE ACCESS and *Telecommunications Systems* (Springer). He has severed in a large number of international conferences in the capacity of the general co-chair, a TPC co-chair, and the symposium chair.