IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 29, 2020, accepted August 7, 2020, date of publication August 11, 2020, date of current version September 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3015856

A Glider-Assist Routing Protocol for Underwater Acoustic Networks With Trajectory Prediction Methods

YISHAN SU[1], (Member, IEEE), LIN ZHANG[1], YUN LI[2], AND XING YAO[2]
[1]Department of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China
[2]Guilin University of Aerospace Technology, Guilin 541004, China

Corresponding author: Xing Yao (yaoxing@guat.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61701335, Grant 61862020, and Grant 61861014.

ABSTRACT In recent years, marine exploration has become one of the most popular research subjects. At present, gliders in Underwater Acoustic Sensor Networks (UASNs) are equipped with a variety of sensors, which can play an important role in marine detection and monitoring. In addition, gliders also have the ability of data collection and storage. When coexisting with the sensor nodes, they can be regarded as special sensor nodes with semi-determined sawtooth trajectories. In hybrid UASNs including sensor nodes and gliders, a robust routing protocol is required to improve the poor network connectivity caused by long transmission delay, high bit error rate and unreliable transmission links. In this article, the Fuzzy Logic Algorithm (FLA) is used to convert multiple input parameters into one output value, thus reducing the storage burden of the network. Therefore, a glider-assist routing scheme with trajectory prediction is proposed to improve the connectivity of the hybrid network. The simulation results show that our scheme is superior to other protocols in terms of delivery ratio, end-to-end latency and lifetime.

INDEX TERMS Hybrid underwater acoustic sensor network, routing protocol, fuzzy logic algorithm, Kalman filtering.

I. INTRODUCTION

The planet we live on is blessed with a large amount of water resources, which is rich in minerals, gas, oil etc., to be explored. Recently, with the depletion of land resources, marine exploration has become one of the most popular research subjects. We can establish a Underwater Acoustic Sensor Networks (UASNs) in a specific area to better explore the ocean [1]. Compared with the terrestrial wireless communication, the transmission condition of underwater acoustic channels is even worse, because the acoustic signals travel five orders of magnitude slower than the electromagnetic waves. In addition to the features of intensive multipath-interference, narrow frequency band, and loud noise in underwater environments, the main difficulty in communication between nodes in UASNs is long end-to-end latency and limited energy. Therefore, the existing routing protocols for terrestrial networks cannot be directly used underwater, so it is necessary to have an efficient routing protocol which can work efficiently in UASNs [2].

At present, with the exception of sensor nodes, gliders equipped with various sensors can also observe and obtain a lot of information, such as temperature, salinity, and ocean current changes, which play a vital role in marine exploration, resource development, and disasters preventing, etc. [3]. In the hybrid network containing sensor nodes and gliders, many sensor nodes can detect that the environment is passively moving with the flow of water. Sensor nodes that are placed in different depths underwater do not have any autonomous movement capabilities. According to the routing protocol, they need to encapsulate the collected data into packets, and then forward them Hop-by-Hop (HH) to the floating sink nodes. Therefore, in UASNs that only contains sensor nodes, the connectivity of the network is not ideal due to the different density of nodes in the networks. In those areas where nodes are relatively dense, a hot node is often used as a forwarding node, resulting in excessive energy consumption. While, in those areas where nodes are relatively

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

sparse, there may be too few neighboring nodes around a certain node, resulting in the emergence of routing voids. In order to improve the connectivity and balance the load of the networks, the existence of mobile nodes in the hybrid network is necessary. The main functions of gliders include marine parameter measurement, autonomous gliding motion control, data storage and long-distance transmission, making gliders suitable as a mobile platform for the maintenance of underwater networks. However, the glider moves according to a semi-determined sawtooth trajectory, rather than a random one. Therefore, a routing protocol designed for hybrid networks is needed which can fully consider the motion characteristics of gliders.

In general, in order to make routing decisions, sensor nodes will take the information from themselves and their neighboring nodes, such as the location and remaining energy and so on as input data information. Based on whether to further process the input data information, the existing underwater routing protocols can be divided into two types: one is a routing protocol with raw information; and the other is a routing protocol with processed information. For instance, classic protocols such as VBF [4] and DBR [5] are routing protocols with raw information, and their routing decisions are made directly based on the location or depth information. They have the advantage of being able to make routing decisions, based on a small amount of input data information. However, as the number of input variables increase, it will not only increase the amount of computation, but also increase the load of network, thus increasing the network latency and shortening the lifetime. The protocols with processed information can introduce other algorithms to further process the input data information. In recent years, many researchers have tried to combine intelligent algorithms with routing decision-making processes, and have made some important progress [6]. Such protocols can effectively improve the performance of the network by selecting appropriate algorithms to process the input data information.

In this scheme, we proposed a glider-Assist Routing Protocol (GARP) for UASNs. To begin with, due to the existence of communication between sensors and gliders in hybrid networks, the load of networks is larger than that of ordinary sensor networks. Therefore, in order to make appropriate routing decisions, under the premise of reducing the load of networks, Fuzzy Logic Algorithm (FLA) [7] is introduced when nodes are establishing their routing tables. In addition to the remaining energy and remaining buffer, we also take the link quality at the neighboring nodes into account when selecting an appropriate forwarding node. By fuzzifying and de-fuzzifying the parameters of each node according to a preset Fuzzy logic rule, sensor nodes can make fair and efficient routing decisions without excessive computing. Furthermore, de-fuzzifying can convert multiple input parameters into one output value, and reduce the storage burden of the network.

In hybrid networks, gliders with semi-determined trajectory can achieve the function of store-and-forward during communication [8], [9]. Therefore, using glider as a relay node can help the network deal with the voids by placing the gliders in a location with poor network connectivity. In the process of establishing the routing table, the gliders keep moving in the network. They collect routing tables from passing nodes and deliver them to the Onshore Data Analysis Center (ODAC) when they float to the surface. By analyzing the number and depth of nodes in the routing table, the horizontal trajectory of gliders was redesigned, and then the vertical trajectory of the gliders can be predicted by using Kalman filter algorithm [10]. Finally, the prediction results of the gliders' trajectory are converted into a new entry and added to the routing table of the relevant nodes. During the routing process, the network delivery ratio and connectivity is optimized by determining whether the data packets are forwarded to a glider or another sensor node.

The rest of this article is arranged as follows: In Section II, we summarize the existing UASNs routing protocols. Before going into the details of our scheme, we describe the hybrid network in Section III. After that, the routing table establishing process based on FLA and the gliders trajectory prediction process based on Kalman filter algorithm is expanded in detail in Section IV and Section V respectively. In the end, the simulation results are shown in Section VI, and the conclusion is presented in Section VII.

II. RELATED WORK

In this section, based on the development of routing protocols for UASNs in recent years, a literature review is made.

As for the routing protocols with raw information, the sensor nodes usually make routing decisions based on information such as residual energy, location, and link quality etc. without further processing these data according to other algorithms. In Vector-Based Forwarding (VBF) routing protocol proposed by Xie *et al.* [4], a virtual ''routing pipe'' is created based on the location information, and with the assistance of this pipe, the selected forwarding nodes are always in the best forwarding position, which greatly reduces the transmission delay. However, since the routing process is very dependent on the pipe, if there are few nodes in the pipe, it will cause routing void, resulting in poor delivery ratio of the network. HH-VBF [11] optimized the process of establishing ''routing pipe'' by enabling each node to establish the ''pipe'' on a Hop-by-Hop basis. Although HH-VBF allows network nodes to be selected more evenly, it also generates more computation, and the problem of routing voids still exists. In the Depth-Base Routing (DBR) [5] protocol, only the depth information is needed to make routing decisions, and too many data packets are flowing between nods. Therefore, the delivery ratio was improved. However, the flooding of packets will consume a lot of energy, and the routing voids still exist in sparse networks. In [12] some improvements have been made on the basis of DBR. By introducing A-DBR to adjust the communication range and B-DBR to find alternative paths, the routing voids can be optimized. The introduction of C-DBR can reduce the end-to-end delay, while CA-DBR is proposed for avoiding collision. However, the four parts of this article are not integrated into one scheme.

In order to solve the problem of routing voids, many new routing schemes have been proposed in recent years. Since it does not involve a part of the algorithm that processes the inputs, we still classify it as classical routing protocols. In the Void-Aware Pressure Routing (VAPR) [13] protocol, in addition to depth, hop count and sequence number are introduced to make routing decisions. On this basis, the forwarding direction between two neighboring nodes is determined. Each node keeps only two forwarding directions for routing decision, which greatly avoids the generation of routing voids. A Channel-Aware Routing Protocol (CARP) [14] was proposed by S. Basagni *et al.* When selecting the relay nodes, they considered factors such as previous successful delivery ratio, the link quality, and the buffer space etc. At the end of network initialization process, each node keeps the hop count from itself to the sink node, which is still the basic criterion for selecting forwarding nodes. Therefore, the delay of transmission is also reduced. However, the PING-PONG mechanism in CARP introduces a number of control packets that will shorten the lifetime of the network. The Distance-Vector-based Opportunistic Routing (DVOR) [15] uses the query mechanism to establish the distance vector between neighboring nodes, which included the next hop node and the hop counts to the sink node. In DVOR protocol, the forwarding node is determined by the distance vector contained in the data packet. In other words, the selection of relay nodes and data forwarding is performed simultaneously. Therefore, DVOR reduces the energy consumption of the network while avoiding overall routing.

In the routing process, the protocols with raw information have the advantage of less computation but when considering multiple inputs, long control packets that are not related to data transmission will be introduced into the network, which will adversely affect the lifetime of the network. In the information processing protocol, some algorithms are introduced into the underwater routing algorithms to process multiple inputs, which improves the overall network performance.

In FLA, an output is obtained through the process of fuzzification and defuzzification of multiple inputs. Therefore, some UASNs routing protocols are integrated FLA into the routing decision process. In the Energy-Efficient Cooperative Opportunistic Routing Protocol (EECOR) [16], energy consumption and packet delivery ratio are considered in optimal relay selection, and a holding timer is also introduced to schedule the packets' transmission. Thus, collisions in the network are effectively reduced, but the routing voids and network lifetime issues remain unresolved because location information is not considered in the routing process. The RRAHB [17] proposed by Chen *et al.* in 2018 also utilized fuzzy decision algorithm for relay nodes selection. Since RRAHB took depth into account when making fuzzy decision, and adopted a priority-based traffic scheduling mechanism to improve the hotpots issues. Based on the fitness function in VBF, MFPR [18] identified the optimal relay node by introducing a flower pollination algorithm to switch between routing cover sets. MFPR has achieved high delivery ratio and less end-to-end latency by considering the quality of service and two-hop-neighbor connectivity of the networks, but it still requires the routing protocols with less inputs. In the adaptive Deep Q-Network-based energy- and latency-aware routing protocol (DQELR) [19], Machine Learning (ML) is introduced into UASNs routing process. Taking the residual energy and depth information of nodes as inputs, the reward matrix Q is calculated as the basis of forwarding nodes selection. However, the training process of the network cannot be performed entirely underwater, so part of the computation in DQELR needs to be performed on land. In addition, the problem of routing voids remains unsolved.

Based on the former research, most routing protocols aimed at improving network connectivity are at the cost of increasing network latency or introducing more input parameters. Therefore, we need to use a routing method that does not affect the latency or input. In the Autonomous Underwater Routing Protocol (AURP) [20], Autonomous Underwater Vehicles (AUVs) are introduced in this routing process. By establishing different data transmission links among sensor nodes, AUVs, and destination nodes, the connectivity of the network is effectively improved. However, in the AURP, the node can only send data packets to the AUV when it moves into the communication range, which is an uncertain process. In recent years, some researchers have introduced AUV into data collection. In [21], the coordinates of sensors can be marked through AUV. In the process of data collection, by selecting Cluster Heads and designing an optimal path for AUV to pass through all CHs, an efficient data collection algorithm was proposed. In [22], both sensor nodes and AUV know their own location, the problem of uneven energy consumption in UASN is alleviated by selecting gateway nodes. In addition, the malfunction probability of AUV is also considered to improve the robustness of the scheme. Compared with AUV, gliders with semi-determined sawtooth trajectories are more suitable for assisting the routing process of the network. Therefore, the gliders with predictable trajectories are introduced into the UASNs system to solve the problem of routing voids and improve the network connectivity.

III. NETWORK DESCRIPTION

The hybrid network proposed in this article includes multiple static sensor nodes and gliders. The sensor nodes are evenly distributed at different depths underwater, and among them there is only one sink node located on the surface of the water [23]. In addition, sensor nodes can obtain their own depth in water. Under the influence of ocean currents, the position of the nodes will slightly change in the horizontal direction, but not in the vertical direction. Therefore, it is believed that the overall connection of the network remains relatively stable [24].

Under the premise of determined horizontal trajectory, gliders equipped with satellite communication equipment can move up and down in a vertical plane at a constant speed of v, and follow the semi-determined sawtooth trajectory [9]. When the gliders rise to the surface, they extend the antennas out of the water, so that they can communicate with the sink node and the ODAC through electromagnetic waves. Moreover, gliders can act as special nodes that communicate with sensor nodes when moving underwater. In addition, because gliders have the function of storing information, the received data packets can be stored in gliders for a period of time before forwarding, so both the sensor nodes and gliders can obtain underwater depth information.

Since the dynamic network is composed of sensor nodes and gliders, the routing scheme proposed in this article is also divided into two parts. Firstly, the routing table of sensor nodes is established by using the FLA. In this process, the gliders are not involved in the computation. After establishing the routing table, the gliders collect the routing table of each node during the movement, which can be used as the basis for replanning the horizontal trajectory of gliders. Compared with the sensor nodes, the position of gliders changes more frequently. Once the glider's vertical trajectory is determined, based on the periodicity of the gliders' motion, the node will be able to predict when the next glider will move into the communication range. Therefore, this node can use the trajectory prediction process based on Kalman filter algorithm to predict the vertical trajectory of the glider. Finally, after going through these two processes, it generates a routing table containing the passing time of gliders.

IV. FLA-BASED ROUTING TABLE ESTABLISHMENT

A. OVERVIEW

In 1965, the concept of "Fuzzy Sets" was first proposed by American mathematician L. Zadeh, which marked the birth of fuzzy mathematics [7]. Based on multi-valued logic, fuzzy logic uses the concept of membership function to distinguish "Fuzzy Sets" and handle fuzzy relationships, which is suitable for qualitative problems with unclear boundaries.

During the routing process, in order to select a appropriate relay among the neighboring nodes to forward the data packets, the nodes need to be evaluated from multiple perspectives. In our scheme, nodes are evaluated from three aspects: the remaining energy (E), the remaining buffer space (B), and the link quality based on Signal-to-Interference-plus-Noise Ratio (SINR). Three membership functions are determined respectively. After running the fuzzy logic decision-making algorithm and the defuzzification process, each node obtains an output as a representative of itself. By running the FLA, three parameters are simplified into one output, which greatly reduces the network overhead and saves energy in the process of routing table establishment.

B. FUZZIFICATION

1) MEMBERSHIP FUNCTION

In fuzzy logic, unlike Boolean Logic, a certain value of a variable does not completely attach to a certain class, but is measured by the degree of membership. Membership function provides us with a measure to determine the degree of membership. The fuzzy logic used in this article has three input variables: Energy, Buffer, and SINR. Each node knows its remaining energy and remaining buffer space. The link quality of a node is measured by the SINR, and can be calculated as follows:

In UASNs, the Thorp propagation model is used to describe the characteristics of underwater acoustic channels in the network. The path loss of underwater acoustic communication channel is not only related to the transmission distance (l), but also to the signal frequency (f). According to the factors that affect the performance of acoustic signal, the attenuation model of acoustic channel is established [25]:

$$A(l,f) = l^k \alpha(f)^l \tag{1}$$

The value of k is 1.5, which is the spreading factor describing the practical spreading, and $\alpha(f)$ is the absorption coefficient measured in dB, which can be defined as follows:

$$10log\alpha(f) = 0.11\frac{f^2}{1+f^2} + 44\frac{f^2}{4100+f} + 2.75 \cdot 10^{-4}f^2 + 0.003 \tag{2}$$

The noise in UASNs is measured in dB, which consists of turbulence noise N_t:

$$10log\, N_t(f) = 17 - 30log\, f \tag{3}$$

distant shipping noise N_s:

$$10log\, N_s(f) = 40 + 20(s - 0.5) + 26log\, f - 60log(f + 0.03) \tag{4}$$

in which s is the shipping activity factor values between 0 and 1. Noise caused by wind-driven waves N_w:

$$10logN_w(f) = 50 + 7.5w^{1/2} + 20log\, f - 40log(f + 0.4) \tag{5}$$

where w is the wind speed in m/s. And thermal noise N_{th}:

$$10logN_{th}(f) = -15 + 20log\, f \tag{6}$$

Thus, the overall noise of UASNs can be derived:

$$N(f) = N_t(f) + N_s(f) + N_w(f) + N_{th}(f) \tag{7}$$

Furthermore, SINR of nodes in UASNs can be derived by:

$$SINR(l,f) = \frac{P/A(l,f)}{N(f)\Delta f} \tag{8}$$

where P is the transmission power, and Δf is the noise bandwidth (a narrow band around the frequency f) of the receiver. So far, the SINR values at sensor nodes can be calculated.

In OFDM modem, each sensor node has 5 operating modes as shown in Fig. 1. Each mode corresponds to different modulation schemes and transmission rates, as shown in Table 1. As we have calculated the SINR at each node, the corresponding working mode is selected accordingly. In addition, in OFDM modem, a data packet is composed of multiple

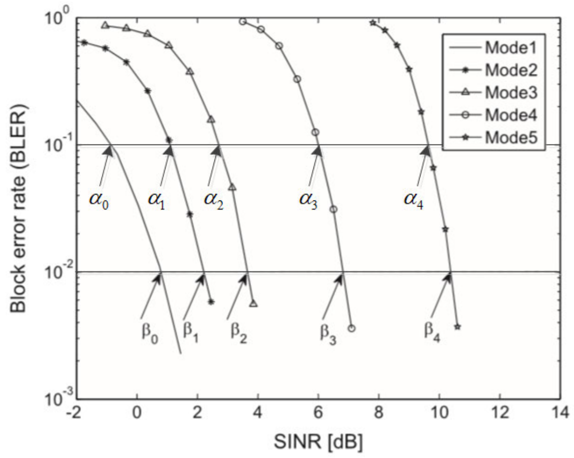

FIGURE 1. BLER for five transmission modes of the OFDM Modem [26].

TABLE 1. Transmission modes in OFDM modem [26].

Mode	Modualtion	Payload per block(bytes)	Rate(kb/s)
Mode 1	BPSK	38	1.38
Mode 2	QPSK	80	2.90
Mode 3	QPSK	122	4.42
Mode 4	16QAM	164	5.94
Mode 5	16QAM	248	8.99

blocks, therefore, Fig. 1 shows the relationship between block error rate (BLER) and SINR in different operating modes. When the BLER exceeds 0.1, the corresponding SINR is set to Poor, and when the BLER is less than 0.01, the corresponding SINR is set to Good. In addition, the remaining energy and remaining buffer of the nodes are converted into percentages. Therefore, according to the rules in Table 2, the membership functions based on Trapezoidal and Triangular are constructed. We have listed the membership functions of energy and output as examples in Fig. 2 Similarly, buffer and SINR have similarly shaped member functions as energy.

TABLE 2. Rules for membership functions.

Level	Energy(E)	Buffer(B)	SINR
Poor	0∼25%	0∼25%	$< \alpha_i$
Average	25%∼75%	25%∼75%	$\alpha_i \sim \beta_i$
Good	75%∼100%	75%∼100%	$> \beta_i$

Here, we give an example to show the computing process of fuzzy logic output value intuitively. It is assumed that the remaining energy of this node is 56.25%, the remaining buffer is 68.75%, and the SINR is 0. Then the degree of membership is determined, according to the membership functions:

$$E(Poor) = 0; E(Avg.) = 0.75; E(Good) = 0.25$$
$$B(Poor) = 0; B(Avg.) = 0.25; B(Good) = 0.75$$
$$S(Poor) = 0; S(Avg.) = 1; S(Good) = 0 \tag{9}$$

2) FUZZY LOGIC DECISION

In this article, the output is in the interval of [0,1], which is also divided into three levels: poor, average, and good.

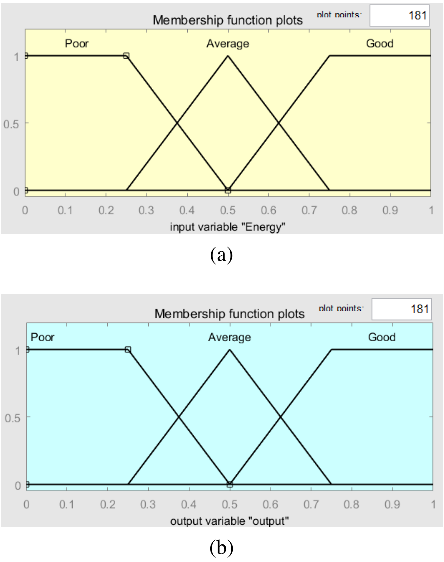

FIGURE 2. (a) Membership function for energy; (b) Membership function for output.

After determining the degree of membership of the input variables, fuzzy logic decision is made according to fuzzy decision rules. We have performed an AND operation on the three variables, that is, taking the minimum value among the three variables and defining the operation result as Fire Strength (FS). For the three inputs, there are 27 logical combinations, and some fuzzy logical decision rules are exemplified in Algorithm 1.

Algorithm 1 Fuzzy Logical Decision Rules

if (E(Poor)), **then** (FS(Poor));
else if (E(Good)**&**B(Good)), **then** (FS(Good));
else if (E(Good)**&**B(Avg.)**&**S(Good)), **then** (FS(Good));
else if (E(Avg.)**&**B(Good)**&**S(Good)), **then** (FS(Good));
else (FS(Avg.)).

When designing the UASNs routing protocols, energy-efficiency is the most important consideration. If a node is frequent used as a forwarding node, then the life of the entire network is likely to end early due to the exhaustion of nodes' energy. Therefore, in the decision rules, as long as the remaining energy of a certain node is at a poor level, the corresponding FS must also at a poor level. In this way, the excessive energy consumption of a single node is reduced, and the lifetime of the network is prolonged.

The fuzzy model surface is shown in Fig. 3

Then, according to the fuzzy logical decision rules, the corresponding FSs can be calculated as follows. At this point, the fuzzification process is complete.

$$FS_1(Avg.) = E(Avg.) \& B(Avg.) \& S(Avg.) = min(0.75, 0.25, 1) = 0.25 \tag{10}$$
$$FS_2(Avg.) = E(Avg.) \& B(Good) \& S(Avg.) = min(0.75, 0.75, 1) = 0.75 \tag{11}$$

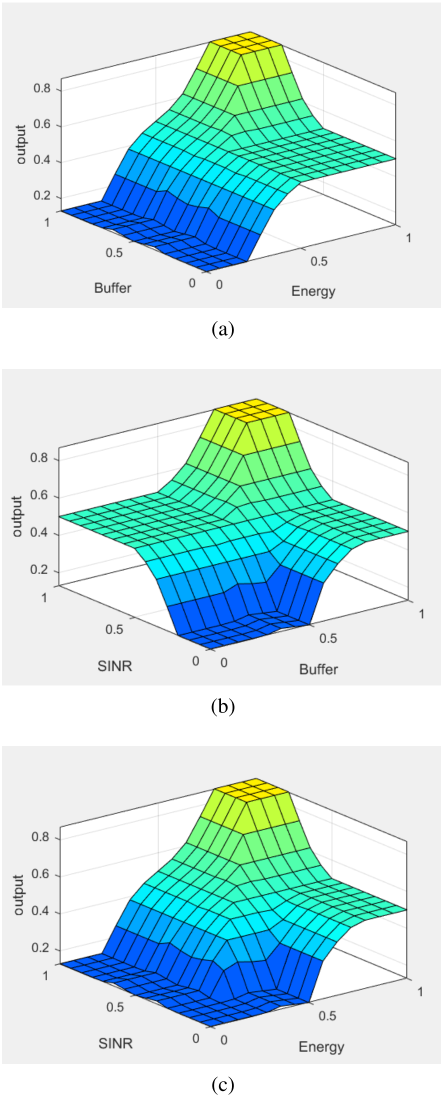

FIGURE 3. (a) Fuzzy model surface for energy and buffer; (b) Fuzzy model surface for buffer and SINR; (c) Fuzzy Model surface for energy and SINR.

$$FS_3(Avg.) = E(Good)\&B(Avg.)\&S(Avg.) = min(0.25, 0.25, 1) = 0.25 \tag{12}$$

$$FS_4(Good) = E(Good)\&B(Good)\&S(Avg.) = min(0.25, 0.75, 1) = 0.25 \tag{13}$$

C. DEFUZZIFICATION

According to the above process, the corresponding FSs are obtained, but they cannot be directly used. Therefore, we need to perform a defuzzification process to obtain a certain output based on FSs. For all FSs, the corresponding weight is added as a weighted average, which is the output value. The weight of each FS is the middle value of the set of each level, which is determined by the membership function of output.

$$W(Poor) = 0.25;\ W(Avg.) = 0.5;\ W(Good) = 0.75 \tag{14}$$

And now we're just going to calculate the final output in here.

$$output = \frac{\sum_i FS_i * W_i}{\sum_i FS_i} \tag{15}$$

$$output = \frac{0.25*0.5 + 0.75*0.5 + 0.25*0.5 + 0.25*0.75}{0.25 + 0.75 + 0.25 + 0.25} = 0.542 \tag{16}$$

Therefore, the final output of the given example is 0.542.

D. ROUTING TABLE ESTABLISHMENT AND UPDATE

After calculating the final output, nodes will broadcast a data packet containing their own depth and defuzzification output. In order to reduce network load and improve transmission efficiency, the number of forwarding the broadcast packets is set to once, i.e., the node does not forward the data packet after receiving it. Then each node in the network will build a routing table (Table 3) based on the received data packets, containing the depth and output information of all its neighbors. Once the routing table is established, it is stored on the node.

TABLE 3. Formats for routing table at nodes.

No.	depth	output
1	d_1	o_1
2	d_2	o_2
...	...	...

Due to the limited energy of underwater nodes, the update of the node's routing table does not need to use the data sent by the neighboring nodes, but its update timing is determined by the neighboring nodes. With the operation of the network, when the remaining energy, the remaining buffer, or the SINR of nodes in the network reached the critical values (25%, 75% or α_i, β_i), the output is recalculated and a new packet for the routing table update is generated. In this way, the routing table is updated in an energy-efficient manner. After the routing table is established, the initialization phase of the network will end. At this time, most nodes of the network can communicate with the sink node according to the routing table.

V. A METHOD OF TRAJECTORY PREDICTION BASED ON KALMAN FILTER ALGORITHM

A. HORIZONTAL TRAJECTORY DETERMINATION

This article only considers the gliders' movement in a vertical plane. The horizontal trajectory is a straight line which can be determined before dropping the glider into the water, and they will follow the sawtooth trajectories in the vertical direction, as shown in Fig. 5.

At the beginning of the deployment of sensor network, the nodes are placed in a known initial position. After that, the routing table establishment process begins. If a node does not receive any data packets from nodes other than those in the routing table in 3 consecutive time slots, it indicates that the routing table of the node is established. The reason

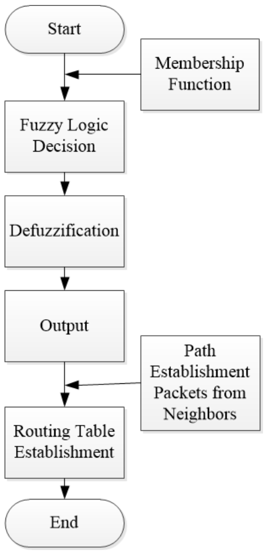

FIGURE 4. Flowchart of routing table establishment.

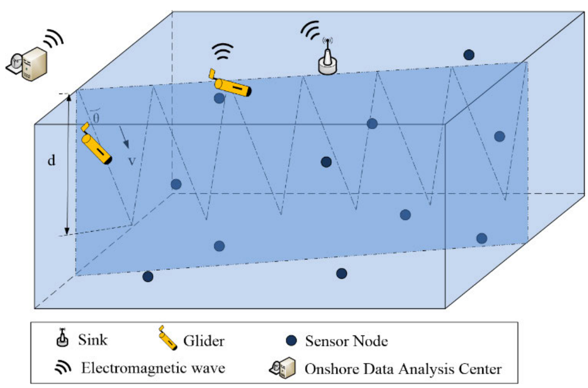

FIGURE 5. Schematic diagram of glider's trajectory.

is that during the routing table establishment process, only path establish packets are transmitted in the network, and the collisions are less likely. In addition, the routing table is established only based on nodes within one-hop communication range, whose transfer time is order of seconds. Therefore, with consideration of the latency of collisions and retransmissions, 3 time slots are enough for nodes to receive path establish packets from nodes within one-hop range.

Then, gliders are dropped into water. Since the initial deployment position of sensors is known, the horizontal two-dimensional coordinates of all nodes are available. Assuming that the coordinates of node i are (x_i, y_i), the glider's horizonal trajectory $y = a + bx$ can be calculated based on the least square method according to the following formula.

$$a = \frac{\sum_i y_i \sum_i x_i^2 - \sum_i x_i \sum_i x_i y_i}{n \sum_i x_i^2 - (\sum_i x_i)^2} \tag{17}$$

$$b = \frac{n \sum_i x_i y_i - \sum_i x_i \sum_i y_i}{n \sum_i x_i^2 - (\sum_i x_i)^2} \tag{18}$$

where n is the amount of the sensor nodes. By changing the nodes involved in the fitting calculation, the horizontal trajectories of multiple gliders can be planned. For example, if two gliders are dropped, their initial trajectory will be planned as the diagonal of the horizontal area in order to cover the monitored water area more. The surface is divided into 9 areas as shown in Fig 6. Excluding areas 1, 9 and 3, 7, two fitting calculations are carried out. Then the initial trajectories of the two gliders could be determined.

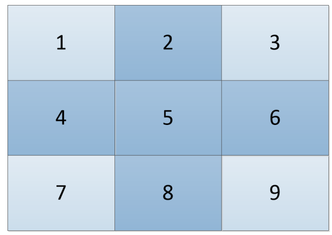

FIGURE 6. Schematic diagram of the water area.

As the routing table of the sensor nodes is completed, gliders begin to collect the routing tables from the sensor nodes within its communication range during their movement. For these packets containing routing tables, only the gliders receive and other nodes in the network do not process it. Every time the glider floats to the surface, it sends the collected information to the ODAC. When the glider does not receive the routing table from new nodes in two consecutive periods T, it means that the routing table collection process is finished. Since the glider can be located by the antenna when it floats to the surface and its trajectory and speed in water are all known, the position of the nodes corresponding to each routing table can be inferred according to [27], [28].

After the two gliders have finished the routing tables collection, further analysis is performed based on the collected routing tables. Nodes that meet the following requirements are selected: those with no more than two neighboring nodes in the routing table and those with only deeper neighbors. By analyzing the locations of these nodes, the horizontal trajectories of the gliders can be re-planned according to Eq(17) (18) so that the gliders can communicate with nodes with poor connectivity. Therefore, the connectivity of the network has been improved.

B. VERTICAL TRAJECTORY PREDICTION

1) KALMAN FILTERING ALGORITHM

Kalman filtering is an algorithm that uses the linear system state equations to optimally estimate the state of a system by observing input and output. Since the observation includes the influence of noise and interference in the system, the optimal estimation can also be regarded as a filtering process. To put it simply, the Kalman filter algorithm is an optimal recursive data processing algorithm. Kalman filtering algorithm is mainly including two parts: predict and update. The main computing process is summarized in Fig. 7. In the prediction part, first, the priori state estimation of the previous

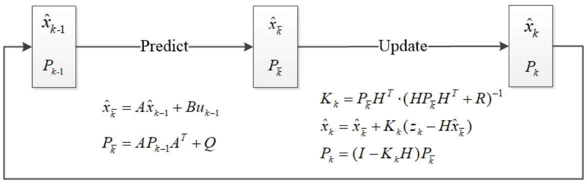

FIGURE 7. Computing process of Kalman filtering algorithm.

time slot $\hat{x}_{k-1}$ and the input u_{k-1} are utilized to calculate the priori state estimation of the current time slot $\hat{x}_{\bar{k}}$. In addition, based on the covariance of priori state estimation of the previous time slot P_{k-1} and the noise of the prediction process Q, the priori estimated covariance of the current time slot $P_{\bar{k}}$ is obtained. After that, the update process begins. First, Kalman gain coefficient K_k is calculated based on $P_{\bar{k}}$ and the observation noise R, which is the core parameter of the Kalman filtering process. Then, the posterior state estimate of the current time slot $\hat{x}_k$ is calculated based on K_k, the observation value z_k and $\hat{x}_{\bar{k}}$. It is the optimal estimated value, and one of the filtering results. In addition, based on K_k, the posterior estimated covariance P_k is calculated, which is another filtering result and also the basis for judging whether the Kalman filtering process is completed. For other details, please refer to [10].

2) VERTICAL TRAJECTORY PREDICTION

In a vertical plane, gliders move at a constant speed of v. As shown in Fig. 8 the angle between the motion track and the vertical direction is θ, and the deepest diving depth is d. The time is divided into time slots k. Both the system noise and measurement follow a Gaussian distribution, and the dynamic system is a linear system. Therefore, Kalman filtering algorithm can be applied in the prediction of glider's trajectory. The calculation process of the Kalman filter is carried out in gliders.

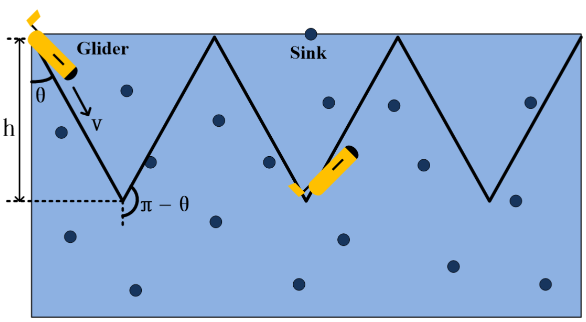

FIGURE 8. Vertical trajectory of glider.

Algorithm 2 summarizes the main processes of vertical trajectory prediction based on Kalman filter algorithm. The terms used in the computing are listed in Table 4.

During a glider is moving underwater, it can measure its depth value z_k at the current time slot k. Firstly, by comparing with z_{k-1}, the measurement at the previous time slot, θ can be

Algorithm 2 Kalman Filter Algorithm Based Vertical Trajectory Prediction

//Initialization: $\hat{x}_0 = 0, z_0 = 0, P_{\bar{k}} = P_k = Inf.$
while ($P_k > Th.$)
// termination condition of Kalman filter algorithm
if ($z_k > z_{k-1}$)
$\theta = \theta;$
else if ($z_k < z_{k-1}$)

$$\begin{aligned} \theta &= \pi - \theta; \\ z_k &= \hat{x}_k + r_k \\ \hat{x}_k &= \hat{x}_{k-1} + v \cdot cos\theta \cdot k + q_{k-1} \\ P_{\bar{k}} &= P_{k-1} + Q \\ K_k &= P_{\bar{k}} \cdot (P_{\bar{k}} + R)^{-1} \\ \hat{x}_k &= \hat{x}_{\bar{k}} + K_k(z_k - \hat{x}_{\bar{k}}) \\ P_k &= (1 - K_k)P_{\bar{k}} \end{aligned}$$

TABLE 4. Definition of terms.

Parameters	Description
k	current time slot
$Th.$	predetermined threshold of covariance
z_k	current state observation of depth
v	speed of glider
θ	the angle between the motion track and the vertical dirction
r_k/R	observation noise
$\hat{x}_k$	the state estimation of depth at k
$\hat{x}_{\bar{k}}$	the prior estimation of depth at k
q_{k-1}/Q	noise of the prediction process, obeys Gaussian Distribution
$P_{\bar{k}}$	covariance of $\hat{x}_{\bar{k}}$
K_k	Kalman gain
t	period of the glider's motion
T	period of the glider to go back and forth within the network

determined. Then, based on the speed v and θ, the estimated depth $\hat{x}_{\bar{k}}$ at the current time slot is calculated. Next, according to the Kalman filter algorithm, the vertical trajectory of the glider is predicted. This process is repeated until the covariance of the state estimation $P_{\bar{k}}$ achieves Cramér-Rao lower bound J_k calculated by Eq(19) [29], which indicates that the trajectory prediction in the vertical direction of the glider has been completed. According to the predicted result, the period of the glider's motion was determined and represented by t.

$$J_k = (Q + J_{k-1})^{-1} + R^{-1} \tag{19}$$

3) ROUTING TABLE UPDATE

As a special node with high dynamics, glider needs to establish a connection with the sensor node by broadcasting path establishment packets containing its motion cycle t. In order to prevent redundancy, the number of forwarding of these data packets is also limited to once, and the node will not forward it after receiving it. The node receiving the packet will add the glider as a new optional forwarding node to its routing table, and record the current time slot k. Then, the time slot of the node's next communication with the glider according to the motion cycle of the glider is calculated.

As shown in Fig 8, the glider makes a round-trip motion throughout the network area. During one round-trip of the glider, it will pass through the same position twice in the form of upward and downward respectively. Supposing the time taken for the glider to go back and forth within the network is T, and $T = nt$. Therefore, after the node recording the time slot k when the glider first passed of it, the next time slots when the glider will pass through this node again can be predicted based on (15) and (16).

$$k_1 = m \cdot T - 2k \tag{20}$$

$$k_2 = m \cdot T + k \tag{21}$$

where m is an integer which updating each time after the glider passes. k_1 corresponds to the time slot of the glider passing through the node when it reaches the boundary and returns. At this time, the glider moves in the opposite direction to k. The updated routing table is shown in Table 5.

TABLE 5. Updated routing table.

No.	depth	output
1	d_1	o_1
2	d_2	o_2
...	...	...
Glider's No.	Floating	Next Time Slot Moving in
G1	True/False	k
G1	False/True	k_1
G1	True/False	k_2
...	...	...

After receiving a data packet, the node first searches its routing table. If there is only glider's information in the routing table and no other nodes are available for forwarding, the node would hold the packet until the glider moving in the next time slot. When the next glider floats to the surface, the package will be forwarded to the sink. In addition, if both the forwarding nodes and glider information available in the routing table, the node first determines whether the glider is in the half cycle of floating. If not, the node will forward the data packet to other nodes according to the routing table; otherwise, the data packet is forwarded to the glider. As a result, the network delivery ratio has been significantly improved.

VI. SIMULATION

In this section, a large number of simulation experiments are carried out on Aqua-sim, which is a NS-2 [30] based simulator for simulating underwater sensor networks. It evaluates the performance of GARP in terms of delivery ratio, end-to-end latency and network lifetime. In order to make the experimental results more intuitive, we normalized the experimental data of each group. The GARP with or without the gliders joined (i.e. fuzzy logic algorithm based routing process) are compared with other three protocols named VAPR [13], DVOR [15] and AURP [20].

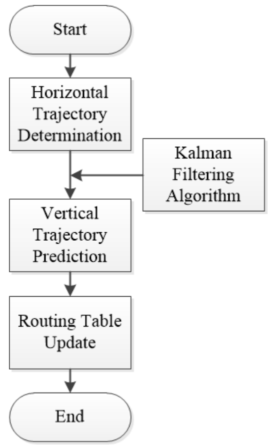

FIGURE 9. Flowchart of gliders' trajectory prediction.

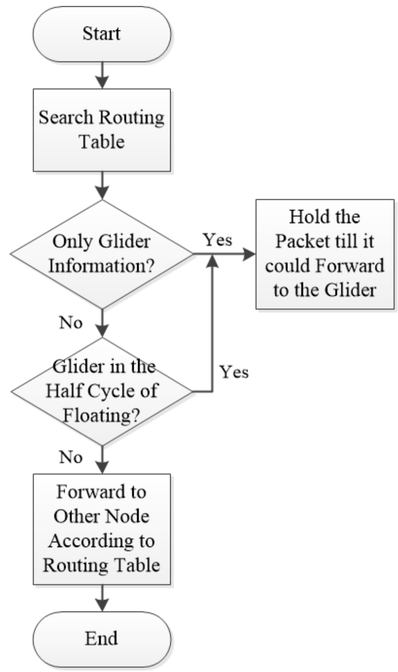

FIGURE 10. Flowchart of routing process.

A. PARAMETER SETTING

The size of simulation area is 5000 $m \times$ 5000 $m \times$ 4000 m, in which two sets of comparative experiments are conducted. Firstly, in order to compare the performance of GARP under different network densities, a set of experiments are conducted. The data packet generation rate is 0.05 packet/s and the number of sensor nodes changes from 50 to 300. Then, we set the number of network nodes to 150, and change the packet generation rate from 0.01 to 0.09 packet/s to do another set of experiments, which can check the performance of GARP in congested networks. In addition, VAPR and DVOR in the comparative experiments are protocols containing only contain sensor nodes. Therefore, in order to ensure the fairness of the comparative experiments, gliders and AUVs in GARP and AURP are used as dynamic sensor nodes. BroadcastMac is utilized as propagation model in the simulations. Other parameter settings are summarized in Table 6 [23].

TABLE 6. Parameter Settinh.

Expermental Parameters	Values
Communication Range	1500 m
Initial Energy	10000 J
Transmitting Power	10 W
Receiving Power	1 W
Idle Power	30 mW
Data Packet Size	150 bytes
Delivery Ratio	10 kbps
Speed of Glider	4.5 km/h
Actual Length of One Time Slot	6 min
θ	45°

B. VERTICAL TRAJECTORY PREDICTION

Fig. 11 shows the predicted vertical trajectory of the glider based on Kalman filtering algorithm. In the simulation experiments, the period of the vertical trajectory of the glider is 2 hours, i.e. 20 time slots. The deepest diving depth of the glider is 3200 meters. It can be seen from the simulation results that the trajectory prediction algorithm based on Kalman filtering algorithm can effectively predict the vertical trajectory of the glider. The predicted results are basically consistent with the glider's movement trend, and can effectively respond to the sudden occurrence during glider's movement. In Fig. 11, when the movement direction of the glider changes, the prediction algorithm can make a corresponding response within 1 time slot. In addition, the error of the prediction result is within 100 meters, which is acceptable compared to the underwater communication range and the speed of the glider, and the impact generated in the route selection process is negligible.

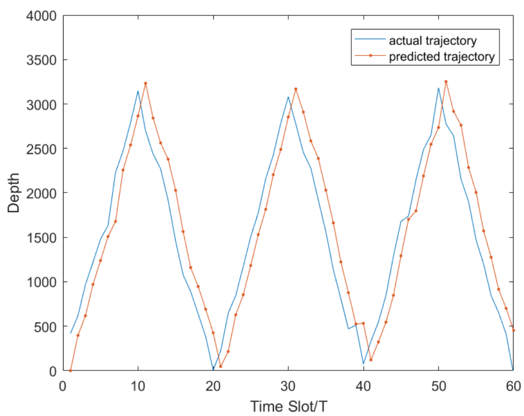

FIGURE 11. predicted vertical trajectory of the glider based on Kalman filtering algorithm.

C. PERFORMANCE EVALUATIONS

1) DELIVERY RATIO

Fig. 12 shows how the delivery ratio varies with the packet generation rate under different protocols in the same network. It can be seen that GARP achieves a higher delivery ratio

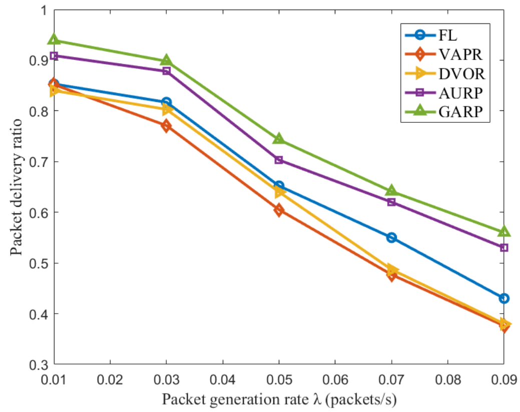

FIGURE 12. Packet delivery ratio comparison among the FL, VAPR, DVOR, AURP and GARP with packet generation rate λ.

than other protocols. In general, the delivery ratio of the three routing algorithms VAPR, DVOR, and FL that do not involve gliders or AUVs, have lower delivery ratios than the other two protocols. Some routing voids in the network can be solved by increasing the hop counts of forwarding routes. However, this method is not suitable to solve all routing void problems. There are still some nodes that cannot be connected to the sink. At this time, the addition of gliders and AUVs can solve this problem more directly and effectively improve the network delivery ratio. In AURP, the trajectory of AUVs is elliptical, and the placement is not targeted. In contrast, GARP can effectively solve the problem of routing voids by deploying gliders in the location with poor network connectivity. In the other three routing protocols, FL unicasts according to the routing table, while VAPR and DOVR both broadcast when sending the data packets. With the increase of packet generation rate, too many redundant data packets will be introduced into broadcasting, and collisions between data packets will lead to the decrease of delivery ratio. However, since DVOR has congestion control mechanism, DVOR is better than VAPR in terms of delivery ratio.

It can be seen that in Fig. 13, the delivery ratio also increases as the number of nodes in the network increases,

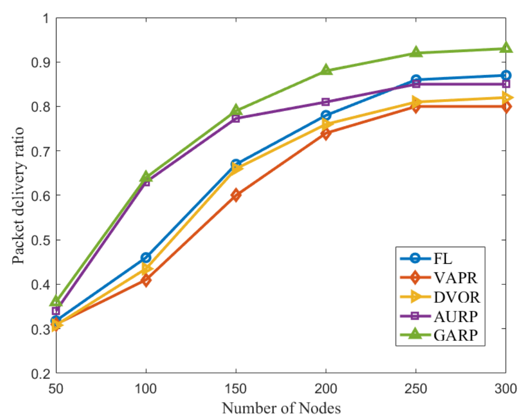

FIGURE 13. Packet delivery ratio comparison among the FL, VAPR, DVOR, AURP and GARP with different network scale.

the delivery ratio of GARP is still higher than other protocols. When the number of nodes in the network is greater than 250, the delivery ratio of AURP will be lower than FL. In the AURP, in order to maintain the connection between sensor nodes and AUVs, they all periodically broadcast beacon packets. Therefore, in a dense network, the probability of packet loss caused by collisions increases, resulting in a lower delivery ratio than FL which unicasts during the routing process.

2) END-TO-END LATENCY

Fig. 14 shows the relationship between end-to-end latency and packet generation rate. It can be seen that the latency of GARP is generally shorter than other protocols, and the latency is less affected by the packet generation rate. In GARP, gliders reduce the number of nodes that originally required multiple hops to reach the sink to one hop, thus greatly reducing the end-to-end latency. In addition, since the glider can also forward data packets in the network, the congestion problem of the network is alleviated with the increase of packet generation rate. Due to the assistance of AUV, AURP has a shorter latency than the other three routing protocols. However, because the trajectory of AUV in AURP is not optimized for routing voids, and with the increase of packet generation rate, a large number of beacon packets will also increase network congestion, thus increasing end-to-end latency. Compared with the flooding packet sending mechanism of VARP, the congestion control mechanism of DVOR can effectively reduce the network load, therefore, the end-to-end latency of VARP is the longest, and the performance of DVOR is similar to FL. But when the packet generation rate exceeds 0.05 p/s, the latency of DVOR increases significantly.

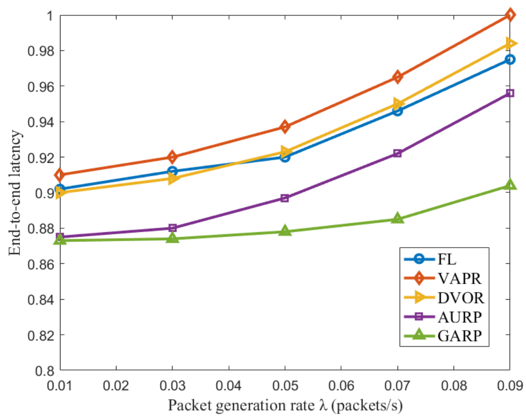

FIGURE 14. End-to-end latency comparison among the FL, VAPR, DVOR, AURP and GARP with packet generation rate λ.

Fig. 15 shows that with the increase of node density in the network, the end-to-end latency shows a downward trend. In general, the latency of GARP is the shortest delay. AURP and GARP still have shorter latency than the other three protocols. In a sparse network, there is a high probability of routing voids, therefore, for routing algorithms without

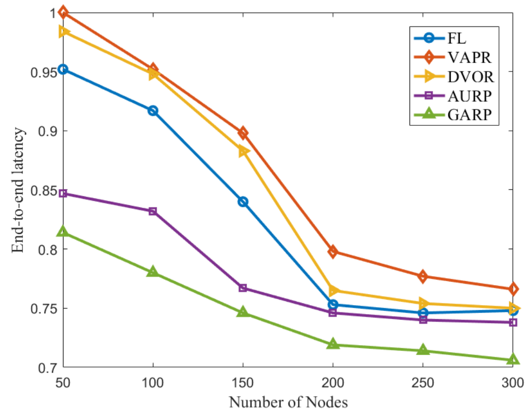

FIGURE 15. End-to-end latency comparison among the FL, VAPR, DVOR, AURP and GARP with different network scale.

gliders or AUV assistance, the network will sacrifice end-to-end latency in exchange for delivery ratio. However, as the increase in the density of nodes in the network can effectively reduce the routing voids in the network, the end-to-end latency also decreases accordingly. However, it can be seen that when the number of nodes in the network is greater than 200, increasing the number of nodes has little impact on the latency of the network.

3) NETWORK LIFETIME

Fig. 16 shows the trend of network lifetime with the increase of packet generation rate. It can be seen that with the assistance of gliders and the Kalman filtering based trajectory prediction mechanism, GARP can effectively prolong the lifetime of the network. In general, as the packet generation rate increases, nodes will consume more energy to send data packets at the same time, so the lifetime of the network expectancy tends to decline. In AURP, sensor nodes, gateways and AUVs all need to broadcast beacon packets to establish and maintain the connection, therefore, the transmission of a large number of redundant packets will greatly

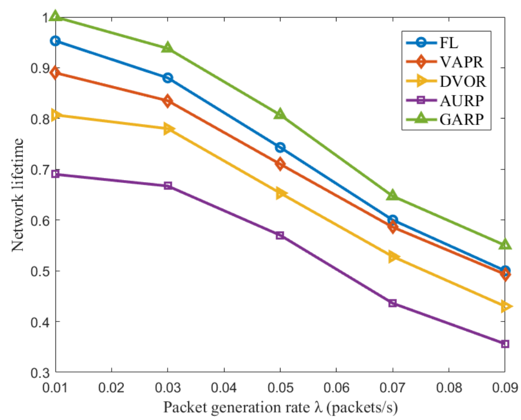

FIGURE 16. Network Lifetime comparison among the FL, VAPR, DVOR, AURP and GARP with packet generation rate λ.

shorten the network lifetime. In addition, since both VAPR and DVOR broadcast data packets, their network lifetime is relatively short. However, when broadcasting data packets, VAPR will decide whether to broadcast upward or downward according to the transmission direction information, so it is more energy-efficient than DVOR and has a relatively longer network lifetime.

Fig. 17 shows that GARP can still maintain a relatively long network lifetime as the number of network nodes increases. Compared with a sparse network, a dense network has a relatively large number of times that a packet is forwarded before reaching the sink node, so nodes in a dense network consume energy faster and have shorter network lifetime. When the number of nodes in the network is less than 100, the network lifetime of VAPR and DVOR is similar. However, when the number of nodes further increases, broadcasting packets significantly reduces the network lifetime of DVOR. In contrast, since FL unicasts data packets, it has a longer network lifetime than VAPR and DVOR.

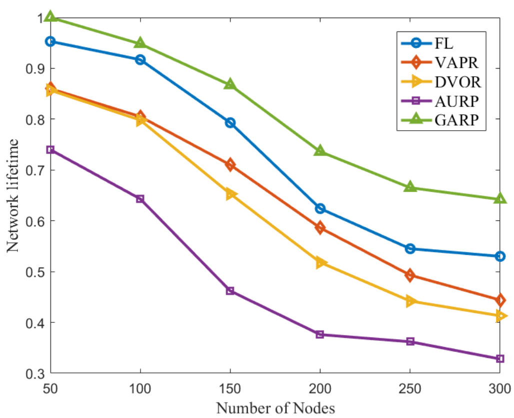

FIGURE 17. Network Lifetime comparison among the FL, VAPR, DVOR, AURP and GARP with different network scale.

VII. CONCLUSION

In this article, we propose a GARP for hybrid UASNs to improve the network connectivity. To begin with, the FLA is applied when sensors making routing decisions. By converting multiple input parameters into one output value, the storage burden of the network is reduced. Then, the horizontal trajectory of the gliders would be re-planned according to the network connection. In addition, Kalman filtering algorithm is utilized to predict the vertical trajectory of the gliders. Therefore, the gliders could work as relays to help the network deal with the routing voids. Simulation results show that compared with other three protocols named VAPR, DVOR and AURP, GARP can effectively improve the delivery ratio and prolong the lifetime of the network.

REFERENCES

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

[2] N. Goyal, M. Dave, and A. K. Verma, "Protocol stack of underwater wireless sensor network: Classical approaches and new trends," *Wireless Pers. Commun.*, vol. 104, no. 3, pp. 995–1022, Feb. 2019.

[3] H. Ghafoor and Y. Noh, "An overview of next-generation underwater target detection and tracking: An integrated underwater architecture," *IEEE Access*, vol. 7, pp. 98841–98853, 2019.

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

[5] H. Yan, Z. J. Shi, and J. Cui, "DBR: Depth-based routing for underwater sensor networks," *Lect. Notes Comput. Sci.*, vol. 4982, pp. 72–86, 2008.

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

[7] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, Apr. 1988.

[8] D. Chen, D. Li, J. Cao, Z. Zeng, B. Yao, and L. Lian, "Dynamic modeling and motion analysis of deep-sea gliders with energy accumulator," in *Proc. OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, May 2018, pp. 1–7.

[9] L. Wang, Y. Li, Y. Liao, K. Pan, and W. Zhang, "Course control of unmanned wave glider with heading information fusion," *IEEE Trans. Ind. Electron.*, vol. 66, no. 10, pp. 7997–8007, Oct. 2019.

[10] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[11] N. Nicolaou, A. See, P. Xie, J.-H. Cui, and D. Maggiorini, "Improving the robustness of location-based routing for underwater sensor networks," in *Proc. OCEANS-Eur.*, Jun. 2007, pp. 1–6.

[12] A. Sher, A. Khan, N. Javaid, S. Ahmed, M. Aalsalem, and W. Khan, "Void hole avoidance for reliable data delivery in IoT enabled underwater wireless sensor networks," *Sensors*, vol. 18, no. 10, p. 3271, Sep. 2018.

[13] 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.

[14] S. Basagni, C. Petrioli, R. Petroccia, and D. Spaccini, "CARP: A channel-aware routing protocol for underwater acoustic wireless networks," *Ad Hoc Netw.*, vol. 34, pp. 92–104, Nov. 2015.

[15] 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.

[16] M. A. Rahman, Y. Lee, and I. Koo, "EECOR: An energy-efficient cooperative opportunistic routing protocol for underwater acoustic sensor networks," *IEEE Access*, vol. 5, pp. 14119–14132, 2017.

[17] C. Qiuli, X. Wei, D. Fei, and H. Ming, "A reliable routing protocol against hotspots and burst for UASN-based fog systems," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 8, pp. 3109–3121, Aug. 2019.

[18] V. Sivakumar and D. Rekha, "A QoS-aware energy-efficient memetic flower pollination routing protocol for underwater acoustic sensor network," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 4, p. e5166, 2020.

[19] 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.

[20] S. Yoon, A. K. Azad, H. Oh, and S. Kim, "AURP: An AUV-aided underwater routing protocol for underwater acoustic sensor networks," *Sensors*, vol. 12, no. 2, pp. 1827–1845, Feb. 2012.

[21] G. Han, X. Long, C. Zhu, M. Guizani, and W. Zhang, "A high-availability data collection scheme based on multi-AUVs for underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 5, pp. 1010–1022, May 2020.

[22] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "An efficient AUV-aided data collection in underwater sensor networks," in *Proc. IEEE 32nd Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, May 2018, pp. 281–288.

[23] J. Liu, W. Guan, G. Han, J.-H. Cui, L. Fiondella, and M. Al-Bzoor, "A dynamic surface gateway placement scheme for mobile underwater networks," *Sensors*, vol. 19, no. 9, p. 1993, Apr. 2019.

[24] Z. Khalfallah, I. Fajjari, N. Aitsaadi, P. Rubin, and G. Pujolle, "A novel 3D underwater WSN deployment strategy for full-coverage and connectivity in rivers," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–7.

[25] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 11, no. 4, p. 34, 2007.

[26] L. Wan, H. Zhou, X. Xu, Y. Huang, S. Zhou, Z. Shi, and J.-H. Cui, "Adaptive modulation and coding for underwater acoustic OFDM," *IEEE J. Ocean. Eng.*, vol. 40, no. 2, pp. 327–336, Apr. 2015.

[27] 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.

[28] Z. Gong, C. Li, and F. Jiang, "AUV-aided joint localization and time synchronization for underwater acoustic sensor networks," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 477–481, Apr. 2018.

[29] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior cramer-rao bounds for discrete-time nonlinear filtering," *IEEE Trans. Signal Process.*, vol. 46, no. 5, pp. 1386–1396, May 1998.

[30] Z. Rahman, F. Hashim, M. Othman, and M. F. A. Rasid, "Reliable and energy efficient routing protocol (REEP) for underwater wireless sensor networks (UWSNs)," in *Proc. IEEE 12th Malaysia Int. Conf. Commun. (MICC)*, Nov. 2015, pp. 24–29.

YISHAN SU (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Tianjin University, Tianjin, China, in 2008, 2010, and 2015, respectively.

He was a Visiting Scholar with the University of Connecticut, CT, USA, from 2012 to 2013. He is currently working as an Assistant Professor with Tianjin University. His research interests include protocols design and implementation in underwater acoustic sensor networks (UASNs) and delay/disruption tolerant networks (DTN).

LIN ZHANG received the B.E. degree in communication engineering from Tianjin University, Tianjin, China, in 2014, where she is currently pursuing the M.S. degree in electronics and communication engineering with the School of Electrical and Information Engineering.

Her research interests include protocols design in underwater acoustic sensor networks (UASNs) and delay/disruption tolerant networks (DTN).

YUN LI received the B.S. degree in electrical engineering and automation from Central South University, China, in 2002, the M.S. degree in communication and information systems from Sun Yat-sen University, China, in 2008, and the Ph.D. degree in information and communications engineering from Tianjin University, China, in 2018.

She is currently a Professor with the Guilin University of Aerospace Technology. Her research interest includes underwater wireless sensor networks.

XING YAO received the B.S. degree in detection guidance and control technology from the North University of China, in 2012, and the M.S. degree in arms engineering from the Beijing Institute of Technology, China, in 2015.

He is currently a Teacher with the Guilin University of Aerospace Technology. His research interests include application of UAV and underwater acoustic communication.