ICST Transactions
on Mobile Communications and Applications

Research Article ICST.ORG

An efficient geo-routing aware MAC protocol for underwater acoustic networks

Yibo Zhu*, Zhong Zhou, Zheng Peng, Michael Zuba, Jun-Hong Cui

Department of Computer Science and Engineering, University of Connecticut, Storrs, CT 06269, USA

Abstract

In this paper, we propose an efficient geo-routing aware MAC protocol (GOAL) for underwater acoustic networks. It smoothly integrates self-adaptation based REQ/REP handshake, geographic cyber carrier sensing, and implicit ACK to perform combined channel reservation and next-hop selection. As a result, it incorporates the advantages of both a geo-routing protocol and a reservation-based medium access control (MAC) protocol. Specifically, with its self-adaptation based REQ/REP, nodes can dynamically detect the best next-hop with low route discovery cost. In addition, through geographic cyber carrier sensing, a node can map its neighbors' time slots for sending/receiving DATA packets to its own time line, which allows the collision among data packets to be greatly reduced. With these features, GOAL outperforms geo-routing protocols coupling with broadcast MAC. Simulation results show that GOAL provides much higher end-to-end reliability with lower energy consumptions than existing Vector-Based Forwarding (VBF) routing with use of a broadcast MAC protocol. Moreover, we develop a theoretical model for the probability of a successful handshake, which coincides well with the simulation results.

Keywords: geographic cyber carrier sensing, geo-routing, MAC, self-adaptation, underwater sensor network

Received on 5 January 2011

doi: 10.4108/icst.trans.mca.2011.e6

1. Introduction

Underwater acoustic network is a promising technique that could connect underwater vehicles, sensor nodes, and other devices working in an underwater environment *via* acoustic channels. It can be used to collect oceanographic data and monitor oceanic volcano activity or oil/gas fields [1–3]. Although it is a class of *ad hoc* networks, the routing and medium access control (MAC) protocols for terrestrial *ad hoc* networks cannot serve it. This is because of its long signal propagation delay, narrow channel bandwidth, and high node mobility. These issues also provide challenges in designing efficient routing and MAC protocols for underwater acoustic networks [1–5].

In underwater acoustic networks, traditional routing protocols such as AODV [6] do not work because their costly route discovery process is unsuitable in long-delay underwater environments. Geo-routing protocols, such as VBF [7], VBVA [8], and DBR [9], are preferred here. These protocols do not need dedicated route discovery and forward packets directly based on the nodes locations. Since location information is indispensable for many aquatic applications [10–14], these protocols do not cause much extra cost and are very efficient from the routing perspective.

However, geo-routing protocols [7, 9] are usually based on the broadcast nature of the underlying acoustic channel. It is highly possible that multiple nodes are selected as the next-hop, which can lead to collisions if all of these next-hop candidates relay the packet. Although the self-adaptation methods such as those in [7, 9] narrow down the size of the candidate set to some extent, the collision probability is still very high without proper MAC design and optimization.

Existing MAC protocols for underwater acoustic networks, such as R-MAC [15], UWAN-MAC [16], and T-Lohi [17], are usually based on channel reservations.

*Corresponding author. Email: yibo.zhu@engr.uconn.edu

In these protocols, senders and receivers interact with each other to reserve the channel for data communications. Before the channel reservation process, sender must know the exact receiver. Unfortunately, it cannot be satisfied by current geo-routing protocols since a node cannot know its next-hop node beforehand in the stateless routing protocol. For example, in R-MAC, a node reserves a channel by measuring the propagation delay and mapping the slot at the sender side to the receiver side, which is not compatible with the geo-routing protocols that cannot provide the next-hop information. Therefore, a new MAC protocol which can effectively suppress collisions and can be smoothly combined with geo-routing protocol is highly desirable.

In this paper, we propose an efficient Geo-rOuting Aware MAC protocoL (GOAL) for underwater acoustic networks which smoothly integrates self-adaptation based REQ/REP, geographic cyber carrier sensing, and implicit ACK to find the next-hop node and perform channel reservation at the receiver side. Utilizing self-adaptation based REQ/REP, a forwarder can determine the best next-hop with little route discovery cost. By adopting geographic cyber carrier sensing, collisions among the data packets are almost eliminated. With implicit ACK strategy, control messages are significantly reduced and thus fewer collisions occur among control packets. With these techniques, GOAL is energy-efficient and provides high end-to-end reliability. Another remarkable feature is that GOAL can work in mobile underwater acoustic networks with localization services such as SLMP [14].

The rest of this paper is organized as follows. Section 2 briefly discusses the related works. Then, GOAL is presented in detail in Section 3. After that, a theoretical model is developed for the probability of a successful handshake in Section 4. Subsequently, simulation results and discussion are shown in Section 5. At last, Section 6 provides our concluding remarks and future work.

2. Background and related work

In this section, we will first review related works on geo-routing protocols in underwater acoustic networks and demonstrate the disadvantages in collision resolutions. We will then review MAC protocols for underwater acoustic networks and their differences from our work.

In underwater acoustic networks, nodes communicate *via* acoustic channels with long propagation delay and therefore take more time and consume more energy to perform route discovery. Luckily, a couple of localization algorithms [11–14] have been designed for underwater acoustic networks, which make geo-routing possible. Geo-routing protocols which are based on the nodes location have been gaining significant attention because they perform stateless routing which allows their routing cost to be very low. In VBF [7], for example, packets are forwarded along the routing pipe from the source to the destination. All nodes within the routing pipe will participate in the packet forwarding process. In DBR [9], a depth-based geo-routing protocol, packets are forwarded to nodes with less depth and finally arrive at the sink nodes deployed on the sea surface. A more recent routing protocol is VBVA [8], which is based on VBF and incorporates the void avoidance capability. In all these protocols, a node does not explicitly choose the next-hop but cooperates with its neighbors to determine the best relay node(s) according to some self-adaptation schemes.

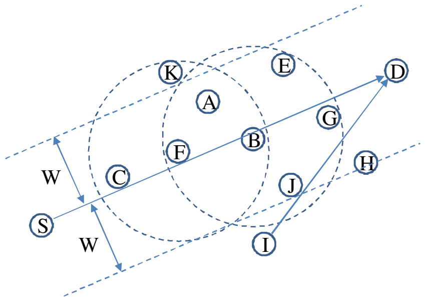

Figure 1. VBF: a self-adaptation based geo-routing protocol.

The basic idea of self-adaptation is as follows. When an eligible node, one that is nearer to the destination than the previous hop, gets a data packet, it starts to back off according to its location before forwarding the packet. The self-adaptation scheme tries to guarantee that a better relay node backs off for a shorter time so that the best relay node ends backoff and then forwards data packet first. For example, as shown in Figure 1, node C and node K are ineligible because the former's position is worse than that of node F and the latter lays out of the pipe area from the source node S to the sink node D. Then, only node A and node B are eligible to forward. Comparing with node A, node B is closer to the vector from source node S to sink node D, and node B is closer to the sink node D than other neighbors of forwarder F. Thus, according to the self-adaptation scheme, the backoff time of node B is shorter than that of node A after receiving the DATA packet from node F. As a result, node B first forwards the DATA packet. By overhearing the forwarding, other nodes, such as node A, cancel the backoff and do not forward the packet any further. In this procedure, several optimal relay nodes[1] can forward first and other nodes are suppressed by the overheard forwarding. As this procedure is repeated, the packet will get closer and finally arrive at the sink node.

Although such a self-adaptation scheme can improve the system's performance to some extent, it cannot prevent MAC collision when there are two or more adjacent nodes forwarding packets at the same time. As shown in Figure 1, if node J happens to forward the DATA packet

[1] Some nodes might have already forwarded the data packet before overhearing the forwarding by the best relay node.

for node I when node B also relays the DATA packet, collision might occur at the common neighbors of nodes B and J, and therefore the DATA packet might not be further forwarded. This harms the end-to-end reliability of the routing protocol. Additionally, the dropped packets waste plenty of energy, i.e. they are not energy-efficient.

In order to further improve the performance of geo-routing protocols, effective collision resolution schemes should be employed, which are usually implemented in the MAC protocols. MAC protocols have been widely investigated for underwater acoustic networks in the last few years. In FAMA [18], RTS/CTS and carrier sensing are combined to avoid collision. However, it is not energy-efficient because the REQ/REP packet is very long and consumes lots of energy to transmit. Slotted-FAMA [19], a modified FAMA, tries to improve the energy-efficiency problem by slotting the time and sending both control and data packets at the beginning of a slot. In this way, the length of an RTS/CTS packet is not determined by the maximal propagation delay as that in FAMA and therefore is much more energy-efficient. However, the RTS/CTS handshake requires the routing protocol to explicitly provide the next-hop, i.e. it cannot act as the MAC protocol for self-adaptation based geo-routing protocols.

COPE-MAC [20], a novel RTS/CTS-based MAC protocol, enables nodes to perform reservations in parallel such that nodes can send multiple DATA packets in one RTS/CTS/DATA/ACK round. For long-delay featured underwater acoustic networks, COPE-MAC is energy-efficient and can significantly improve the throughput. Unfortunately, similar to Slotted-FAMA, it also requires the explicit next-hop and therefore cannot be coupled with self-adaptation based geo-routing protocols.

In T-Lohi [17], a short tone message is used to reserve the channel to send data. However, even though a node does not receive any tone during a contention period, it cannot ensure that there is no collision at the receiver side. In other words, it still suffers from the hidden terminal problem and cannot effectively avoid the collisions.

R-MAC [15] consists of three phases. In the first phase, each node measures the propagation delay to its neighbors. In the second phase, each node reserves a receiving slot at the receiver side and then the receiver confirms if the reservation is collision-free. This phase can make sure that there is no collision at the receiver side for the data packet. In the last phase, each node follows the reservation in the second phase to transmit the data packets. An explicit receiver address is needed in phase two for the channel reservation, and therefore R-MAC cannot work with self-adaptation based geo-routing protocols as well.

Unlike other reservation-based MAC protocols, UWAN-MAC [16] does reservation *via* one-way communication. Assuming the delay between neighbors does not vary, each node piggybacks the relative sending time of the next packet in current packet. As a result, a node knows when it will receive the next packet. However, such a one-way handshake cannot solve the hidden terminal problem. Therefore, collision is still heavy in multi-hop networks. In addition, UWAN-MAC requires nodes to foresee the exact sending time of the next packet, which is unpractical in self-adaptation based geo-routing protocols.

Different from above works, GOAL, the new approach in this paper, smoothly integrates the self-adaptation scheme and MAC reservation techniques. It first employs the self-adaptation scheme to do handshaking and finds the next-hop. Similar to implicitly finding the best relay in self-adaptation based geo-routing protocols, the cost of selecting the next-hop is low in our protocol. Additionally, the receiving slot is reserved based on the geographic information during the handshake, and then the DATA packet can be forwarded without collision. Thus, GOAL finally avoids more collisions while keeping a low routing cost.

3. Description of GOAL

In this section, we will discuss our new GOAL, which is reservation-based and can smoothly integrate with any known geo-routing protocols with self-adaptation. For instance, if GOAL adopts the self-adaptation scheme of VBF, its functionality is equivalent to that of a reservation-based MAC protocol coupling with VBF. We first present the basic idea of GOAL. Then, we describe its three key components, self-adaptation based REQ/REP handshake, geographic cyber carrier sensing, and implicit acknowledgement. Specifically, we apply the self-adaptation scheme of VBF to GOAL as a special case in the description. (Note that GOAL can be used with any self-adaptation scheme.) After that, we will provide an example to show the overall working process of GOAL with detailed analysis.

3.1. Basic idea

Geo-routing aware MAC protocol is a reservation-based MAC protocol. In GOAL, each node maintains a time schedule, which records the time slot corresponding to its neighbors' packet sending/receiving time. Whenever a node wants to send a packet, it should make sure that the selected sending time does not overlap with any existing time slot in the time schedule line. In this way, the DATA packet can be sent collision-free. In order to map the sending/receiving slot to a node's own time schedule line, the self-adaptation based REQ/REP handshake is employed where only a few qualified neighbors are allowed to reply a REP packet for the REQ packet received, which will reduce the collisions. The REQ/REP handshake process in GOAL is used to implement twofold functionalities: determining the next-hop and

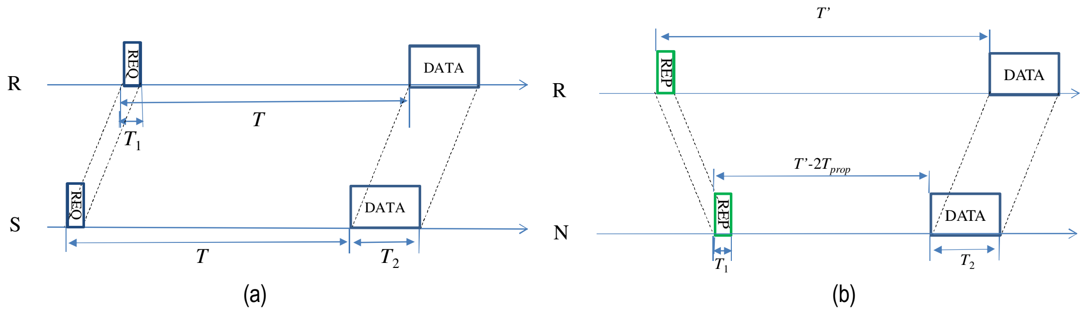

Figure 2. (a) Mapping neighbor's sending slot and (b) mapping neighbor's receiving time slot.

mapping the neighbors' sending and receiving slots to a node's own time schedule.

As shown in Figure 2(a), when sending a packet, node S piggybacks the transmission time T_2 and the relative sending time T of the next packet. After receiving this packet, node R can calculate the receiving time of the packet, i.e. it maps the sending time of the packet at node S to its time schedule line. Specifically, the interval between the time when node S sends the first and second packet is T. Assuming that the propagation delay between nodes R and S does not change much during the coming time period T, the interval between the time points when node R begins to receive these two packets is still T. Therefore, after node R finishes receiving the first packet, it knows that it will receive next packet during time slot $[T - T_1, T - T_1 + T_2]$, where T_1 and T_2 are the transmission times of the first and second packet, respectively. Note that the time slot is expressed by relative time and can be easily converted to absolute time.

The method of mapping the receiving time is illustrated in Figure 2(b). When node R sends the first packet, which has a transmission time of T_1, it then notifies its neighbors including node N that it will receive the next packet after a time period of T'. Suppose that node N knows that the propagation delay between itself and node R is T_{prop}. After completely receiving the first packet, node R knows that there will be a collision at node R if it emits any packet signal during $[T' - T_1 - 2T_{\text{prop}}, T' - T_1 - 2T_{\text{prop}} + T_2]$, where T_2 is the transmission time of the packet which node R will receive. Therefore, in order to avoid the collision at node R, node N must make sure that the sending interval does not overlap with time slot $[T' - T_1 - 2T_{\text{prop}}, T' - T_1 - 2T_{\text{prop}} + T_2]$ when it sends out any packet.

Applying these two mapping schemes, a node can map its neighbor's sending and receiving time period to its own time schedule line to avoid collisions when transmitting a DATA packet.

3.2. The GOAL protocol

The GOAL protocol consists of three parts: self-adaptation based REQ/REP handshake, geographic cyber carrier sensing, and implicit acknowledgement. As described in Section 3.1, self-adaptation based REQ/REP handshake and geographic cyber carrier sensing are used to determine the optimal next-hop and reserve channel for DATA packets. In addition, implicit acknowledgement is imported to reduce the number of control messages. The details of these parts are provided as follows based on the network topology shown in Figure 1.

Self-adaptation based REQ/REP handshake. When the current forwarder F intends to send out a DATA packet, it first selects a qualified sending time to broadcast a REQ$\{P_S, P_F, P_D, T, T_{\text{DATA}}\}$ packet. *Via* the REQ packet, node F tells its neighbors that it will send the DATA packet T time later[2] and the corresponding transmission time is T_{DATA}. It also provides the location of the source, current forwarder, and the destination, which are P_S, P_F, and P_D, respectively.

After receiving REQ, neighbors of node F then know that they will receive the DATA packet $T - T_{\text{REQ}}$ time later (by applying the method of mapping neighbor's sending slot), where T_{REQ} is the transmission time of the REQ packet. Then, the neighbors who have a better location than that of node F start to back off according to the self-adaptation scheme of VBF. Once a node terminates the backoff process, it will send node F a REP$\{P_{\text{thisnode}}, T', T_{\text{DATA}}\}$ packet, where P_{thisnode} is the location of this node and T' is the relative time that it will send a DATA packet. Due to the broadcast feature of acoustic medium, part of node F's neighbors, which might still be in the backoff state, can overhear the REP packet. Then, they cancel the backoff process because the overheard REP implies that there is a better relay. Finally, node F decides the next-hop according to the received REP packets. If it does not receive any REP, it waits for a random time period and tries to resend the REQ. Otherwise node F must receive at least

[2] Note that T must be bigger than double the maximum propagation delay plus the maximum backoff time, which is decided by the self-adaptation scheme. Otherwise, node F cannot determine the next-hop before it sends a DATA packet. In this case, node F will issue a new REQ for this DATA packet later.

one REP packet from its neighbors. In this case, it sets the next-hop as the one with the shortest adaptation time, which node F can for use the location information in REPs to calculate again. Once the pre-scheduled DATA sending time comes, node F sends the DATA packet to the selected next-hop.

As an improvement, multiple DATA packets to the same sink node can be transmitted in one packet train [19], and therefore REQ/REP handshake could perform reservation for multiple DATA packets in one round. This strategy can clearly enhance the efficiency of handshake.

Geographic cyber carrier sensing. In underwater acoustic networks, it is difficult for nodes to completely avoid collisions due to the long propagation delay. In order to address this issue, nodes in GOAL apply the geographic information and the two mapping methods in Section 3.1 to map neighbors' packet sending and receiving slots to their time schedule line. In this way, nodes can figure out when they can send a packet and when they cannot. This is similar to carrier sensing, and therefore we have named it geographic cyber carrier sensing. With geographic cyber carrier sensing, the collisions at neighbors can be greatly alleviated if the selected packet sending time does not overlap with any reserved slot in its time schedule line.

Specifically, after receiving the REQ packet, a node knows that it will receive the DATA packet during $[T - T_{\mathrm{REQ}}, T - T_{\mathrm{REQ}} + T_{\mathrm{DATA}}]$ by applying the method of mapping neighbor's sending time slot, where T_{REQ} stands for the transmission time of REQ packet. Then, this node converts the time slot to absolute time and inserts it into its time schedule line.

The REP packet has twofold functionalities: responding the REQ packet and notifying neighbors to avoid collisions. On one side, with the REP packet from node B, the sender of REQ knows that node B is a potential next-hop. On the other side, based on the information in REP packet, other neighbors of node B can evaluate the propagation delay T_{prop} between themselves and node B. The evaluation method is to use propagation speed to divide the Euclidean distance. Then by applying the method of mapping neighbors' receiving time slot, this node will know that there will be a collision at node B if it sends packet during $[T' - T_{\mathrm{REP}} - 2T_{\mathrm{prop}}, T' - T_{\mathrm{REP}} - 2T_{\mathrm{prop}} + T_2]$, where T_{REP} is the transmission time of REP packet. To avoid collisions, this node should not emit any packet signal during this period. Note that the propagation delay measure method might introduce an error because the acoustic signal is transmitted along a bent path and the nodes are mobile. In order to tolerate this error, guard time T_{guard} is introduced, i.e. the propagation delay is in range $[T_{\mathrm{prop}} - T_{\mathrm{guard}}, T_{\mathrm{prop}} + T_{\mathrm{guard}}]$. Thus, the time period becomes $[T' - T_{\mathrm{REP}} - 2T_{\mathrm{prop}} - 2T_{\mathrm{guard}}, T' - T_{\mathrm{REP}} - 2T_{\mathrm{prop}} + T_{\mathrm{DATA}} + 2T_{\mathrm{guard}}]$.

Based on geographic cyber carrier sensing, nodes can obtain their neighbors' sending and receiving schedule after the REQ/REP handshake. By recording the schedules in their time schedule line, nodes can conveniently choose a qualified time to send packet.

Implicit acknowledgement. In terrestrial *ad hoc* networks, REQ/REP/DATA/ACK can substantially improve the reliability of one-hop transmissions. However, if this scheme is applied in underwater acoustic networks, there are more collisions among control packets because of low bandwidth and long propagation delay. A possible way to address these challenges is to adopt an implicit acknowledgement scheme to reduce the number of control packets. Specifically, if the node that receives the DATA packet is not the destination, it must send REQ to determine the next-hop within a certain time. Because the previous hop is still within the one-hop range with a high probability, it can also overhear the REQ. Based on this heuristic rule, REQ is revised to include the packet identifier (PID) of the DATA packet. As a result, the previous hop can confirm that the DATA packet is successfully forwarded.

For the destination node, it explicitly acknowledges the DATA packet using an ACK packet. In addition, a node will send an explicit ACK packet without a backoff if it receives a REQ when both of the following conditions are met: (i) the location of this node is better than the sender of REQ; (ii) this node has received the DATA packet for which the REQ packet requested.

For any node, if it does not receive an implicit acknowledgement or ACK packet within certain time after sending out the DATA packet, it will initiate a new REQ/REP/DATA round to retransmit the DATA packet. Although retransmission can improve the transmission reliability, the maximum number of times that retransmission can occur should not be infinite. This is because it will introduce more delay and energy consumption. Therefore, we define the number of maximum retransmission times as a tradeoff. Specifically, one node can transmit and retransmit a DATA packet at most the number of maximum retransmission times, where a retransmission is caused by having failed to overhear the corresponding REQ packet or an ACK packet. If the number of maximum retransmission times is exceeded, the node should give up trying to resend the DATA packet.

3.3. An example of GOAL

In the example, the network topology is shown as Figure 3(a). Node F tries to forward the DATA packet from source node S to destination node D. Following the GOAL protocol, node F selects a qualified sending time to broadcast a REQ$\{P_{\mathrm{S}}, P_{\mathrm{F}}, P_{\mathrm{D}}, T, T_{\mathrm{DATA}}, \mathrm{PID}\}$ packet. By means of the REQ packet, node F notifies its neighbors that it will send the DATA packet T time later and the corresponding transmission time is T_{DATA}. With the information in the REQ packet, nodes C, A, and B figure out that they will receive the DATA packet $T - T_{\mathrm{REQ}}$

time later. Note that node C will discard this REQ because its location is worse than that of current forwarder node F. Therefore node *F*s neighbors, except node C, start to back off according to the self-adaptation scheme in VBF.

Similar to VBF, when node B first exits the backoff state, it sends a REP$\{P_B, T', T_{DATA}\}$ packet to node F. By overhearing the REP packet from node B, node A realizes that there is a better relay and cancels the backoff. Additionally, based on the information in the REP packet, node A can evaluate T_{BA}, which denotes the propagation delay between node B and node A. Thus, node A will not send any packet during time interval $[T' - T_{REP} - 2T_{BA} - 2T_{guard}, \ T' - T_{REP} - 2T_{BA} + T_{DATA} + 2T_{guard}]$. As well, node F finds out that the next-hop could be node B after receiving the REP packet. When the scheduled DATA packet sending time comes, node F sends the DATA packet to node B since node B is the optimal one. Later on, node B tries to forward the DATA packet and sends a REQ$\{P_S, P_B, P_D, T, T_{DATA}, \text{PID}\}$ packet to do the handshake. After receiving this REQ packet, node F knows that the forwarding is successful and then prepares to forward the other DATA packets.

3.4. Properties of GOAL

In GOAL, nodes apply the self-adaptation scheme in the REQ/REP handshake process to determine the next-hop. This procedure is similar to the general self-adaptation based geo-routing protocol for data packets. Since a REQ/REP packet is much shorter than a DATA packet, the probability of collision among REQ/REP packets in GOAL is much lower than that among DATA packets in self-adaptation based geo-routing protocol. Note that the use of geographic cyber carrier sensing allows DATA packets in the GOAL protocol to be almost collision-free, and the entire collision probability is accordingly lower than that in self-adaptation based geo-routing protocols. As a result, GOAL provides a higher end-to-end reliability than self-adaptation based geo-routing protocol coupling with broadcast MAC.

As discussed above, GOAL introduces MAC collision among short REQ/REP packets while avoiding collision among long data packets. As a result, the collision probability is reduced. It is clear that the collision among long DATA packets wastes more energy than the collision among short ones, and thus GOAL requires less energy consumption for packet delivery than self-adaptation based geo-routing protocols plus broadcast MAC.

However, in order to achieve the above desirable features, GOAL incurs a longer delay. As explained in Section 3.2, nodes schedule the sending time of a DATA packet *T* time later after sending the REQ packet, where *T* is at least the maximum backoff time plus double the maximum propagation delay. Moreover, due to implicit acknowledgement strategy, nodes will also wait for more than one round trip time. In addition, nodes in GOAL will perform retransmission if any failure occurs during the forwarding procedure, which will also increase the delivery delay. Therefore, the delivery delay in GOAL is higher than self-adaptation based geo-routing protocols plus broadcast MAC.

4. Analysis of probability of successful handshake

In this section, we theoretically explore the probability of a successful handshake in GOAL.

4.1. Notations

In this section, we define the terms in the analysis as follows.

T_I: The given node maps the arrival time points of all neighbors' DATA packets to its own time schedule line. Then, T_I is the length of the interval between the arrival time points of two successive DATA packets on the time schedule line.

T_I': The given node maps the scheduled sending time of all neighbors' DATA packets to its own time schedule line. Then, T_I' is the length of interval between the sending time points of two successive DATA packets on the time schedule line.

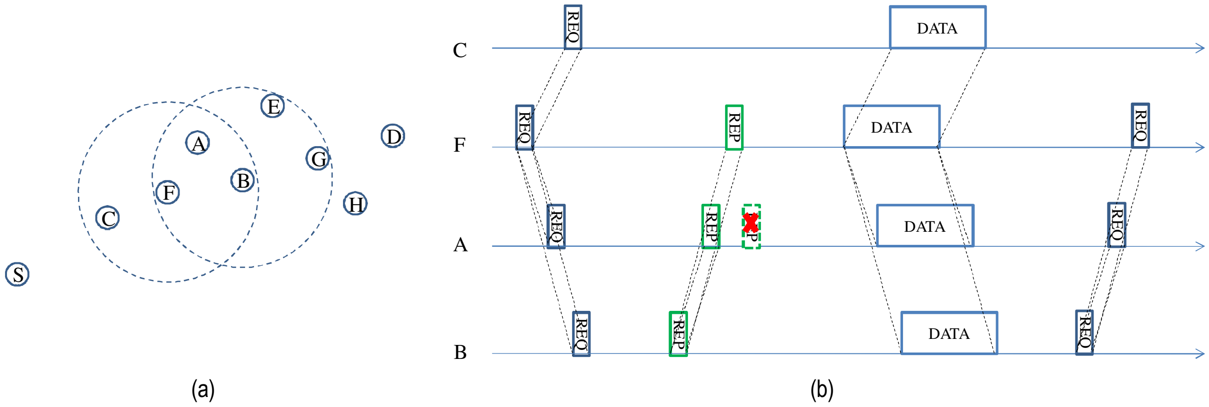

Figure 3. (a) Network topology and (b) one hop forwarding.

T_{D}: Transmission time of the DATA packet.

T_{c}: Transmission time of the control packet (both REQ packet and REP packet).

n: The average number of neighbors of each node.

k: The average number of neighbors which reply a REP packet to the sender of a REQ packet based on the self-adaptation algorithm used by GOAL.

λ: The traffic rate of DATA packets generated by each node.

λ_{D}: The real traffic rate of DATA packets that each node sends to the channel.

P_{c}: The probability that the control packet is transmitted successfully.

P_{r}: The probability that the requested time slot does not overlap with any existing reserved slot on the time schedule line.

$\tilde{P_{\mathrm{H}}}$: The probability of the successful handshake.

4.2. Protocol simplification and assumptions

In GOAL, whenever the MAC layer gets a DATA packet from the upper layer, it will cache this DATA packet if it is forwarding another DATA packet in a REQ/REP/DATA round. After transmitting the current DATA packet successfully, the node starts a new round to transmit the next DATA packet. To simplify the analysis, we revised GOAL such that a REQ/REP/DATA round for different DATA packets can be overlapped. Specifically, when a node is forwarding a DATA packet in a REQ/REP/DATA round, it will cache the DATA packet from the upper layer and send a REQ to perform reservation for the DATA packet in the coming interval between DATA packets. Additionally, retransmissions will no longer happen. In other words, if REQ/REP handshake fails or a DATA packet is transmitted unsuccessfully, the DATA packet will be dropped. We also make the following assumptions throughout the analysis part of this paper.

(i) Similar to existing works [21–23], packet error rate is not considered in the model. In other words, a packet is dropped only if there is a collision.

(ii) The traffic at every node follows a Poisson process with same λ parameter and the DATA packets are of the same length.

(iii) The length of the REQ packet is same as that of the REP packet.

(iv) Relay traffic is not considered, i.e. all nodes only do one-hop transmission.

(v) As described before, GOAL employs implicit ACK and the ACK packet is only sent by the sink node. Without relay traffic, there are few ACK packets in the networks. Therefore, we reasonably ignore ACK packets in the analysis.

(vi) A multi-hop network scenario is considered in the analysis. Accordingly, the analysis also involves the interferences incurred by hidden terminals, which is an important issue but ignored by most of the existing analysis works on MAC protocols.

4.3. Analysis

In this part, we deduce $\tilde{P_{\mathrm{H}}}$ by analyzing the handshake process between sender S and its neighbors $\mathrm{R}_i(1 \leq i \leq n)$, where node S is an arbitrary node in the network.

In GOAL, as illustrated in Figure 4, every node records the time slots when its neighbors receive or send DATA packets. To avoid the collision with DATA packets, control packets (both REQ and REP) are transmitted during the interval between these time slots.

Although the arrival time of the DATA packet follows a Poisson process, the sending time of the DATA packet and the control packet no longer follows the Poisson process due to the schedule strategy in GOAL. However, because every DATA packet is preceded by a REQ packet and the REQ is sent immediately during the next available interval after a DATA packet arrives, the probability that N_{c} control packets are sent by node R_i and its neighbors, except node S, during a given interval of length $T_{\mathrm{I}}^{'}$ equals the probability that N_{c} DATA packets arrive during the same time period. Thus, this probability still can be evaluated by utilizing the Poisson process. Therefore, if $T_{\mathrm{I}}^{'}$ and N_{c} are fixed, we can obtain $P_{N_{\mathrm{c}}}$ as

$$P(N_{\mathrm{c}} = n_{\mathrm{c}}) = \frac{\lambda_{\mathrm{c}}^{n_{\mathrm{c}}} e^{-\lambda_{\mathrm{c}} T_{\mathrm{I}}^{'}}}{n_{\mathrm{c}}!}, \tag{1}$$

where λ_{c} is the traffic rate of control packets sent by node R_i and its neighbors, except node S, total n nodes. Since the traffic of control packets consists of REQ traffic and REP traffic, we can obtain

$$\lambda_{\mathrm{c}} = n(\lambda_{\mathrm{REQ}} + \lambda_{\mathrm{REP}}). \tag{2}$$

Note that every DATA packet is preceded by a REQ packet, which implies REQ packets are subject to the same Poisson process as the input traffic. Therefore, we have $\lambda_{\mathrm{REQ}} = \lambda$. Regarding λ_{REP}, it is related to λ_{REQ}. Specifically, due to collisions, a REQ packet correctly arrives at node R_i with probability P_{c}. After getting the REQ packet, node R_i checks if the requested time slot overlaps with any existing reserved time slot for DATA packets. If there is no overlap, it will reply a REP packet. Otherwise, it will discard the packet. According to the self-adaptation scheme, k out of the n neighbors of node S will reply the REP packet on average, i.e. each neighbor replies with probability k/n. Additionally, considering that each node is a potential REP replier of its n neighbors, we can get

$$\lambda_{\mathrm{REP}} = n\frac{k}{n}P_{\mathrm{c}}P_{\mathrm{r}}\lambda_{\mathrm{REQ}} = kP_{\mathrm{c}}P_{\mathrm{r}}\lambda_{\mathrm{REQ}}, \tag{3}$$

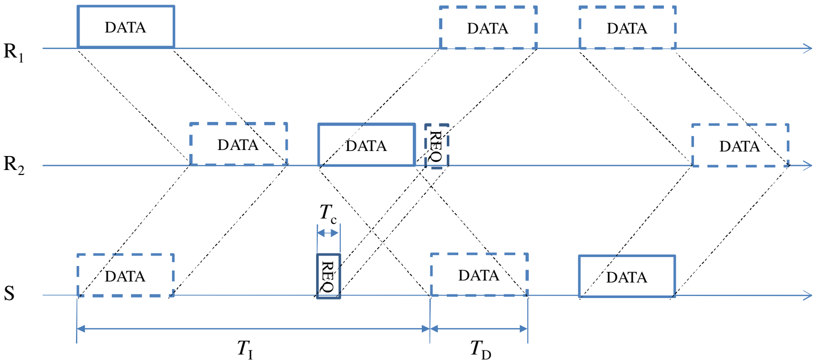

Figure 4. An example of reserved time slots on nodes' time schedule line.

where P_r is the probability that the requested time slot does not overlap with any existing time slot for DATA packets. It can be evaluated as

$$P_r = e^{-2T_D n\lambda}. \tag{4}$$

Substituting equations (4) and (3) into equation (2), we have

$$\lambda_c = n(1 + kP_c P_r)\lambda_{REQ}. \tag{5}$$

Based on P_{N_c}, we can calculate the probability of event A, which denotes the event that a REQ packet sent by node S collides with any control packets at node R$_i$. According to the strategy in GOAL, the sending time of control packets approximately follows a uniform distribution. If there are N_c control packets sent or received by R$_i$ in the interval of length T'_I, excluding the control packets from node S because there is no collision among the control packets sent by node S, then the collision probability of event A is

$$P\left(A|N_c = n_c, T'_I = T\right) = \left(1 - \frac{2T_c}{T - T_D}\right)^{n_c}. \tag{6}$$

Note that the handshake is successful as long as node S receives a REP packet from any neighbor. Recall that on average there are k potential repliers according to the self-adaptation scheme. Accordingly, we can calculate the probability of the event H, which represents the successful handshake, *via* equation (7).

$$P(H|N_c = n_c, T'_I = T) = 1 - (1 - P(A|N_c = n_c, T'_I = T))^k. \tag{7}$$

Then, considering the distribution of N_c, we can get

$$P\left(H|T'_I = T\right) = \sum_{n_c=0}^{\infty}\left(1 - \left(1 - P\left(A|N_c = n_c, T'_I = T\right)\right)^k\right). \tag{8}$$

From equation (8), we can see that we still need to figure out the distribution of T'_I to get $\tilde{P}_H$. Although T_I follows an exponential distribution, T'_I does not follow the exponential distribution anymore due to the schedule strategy in GOAL. As shown in Figure 5, for example, the i^{th} packet arrives at t_i, and the duration of interval $[t_i, t_{i+1}]$ is T_i. After being scheduled, the sending time of the i^{th} packet becomes t'_i, and the interval between t'_i and t'_{i+1} is T'_i. During this procedure, the interval length between DATA packets is adjusted to be at least $T_m(T_m \geq T_D)$.

For a given interval sequence $T_i(0 \leq i \leq l+1)$, there are two conditions for that l successive intervals $T_1 \sim T_l$ are adjusted to T_m but T_{l+1} is reduced to an interval longer than T_m. One is that T_0 should be long enough to allow $t'_1 = t_1$. Here, $t'_1 = t_1$ means the scheduled time t'_1 is not affected by schedule scheme in GOAL. And the other is

$$\begin{array}{l} \sum_{i=1}^{j} T_i \leq jT_m \quad \text{for } 1 \leq j \leq l \\ \sum_{i=1}^{l+1} T_i > (l+1)T_m \end{array}. \tag{9}$$

Note that all T_i follow an exponential distribution with the same λ_{dc} parameter, where λ_{dc} is the total traffic rate of DATA packets sent by any given node, its neighbors, and hidden terminals. Since the time slots confirmed by REP packets might overlap with the slots requested by REQ packets, some slots are not successfully reserved. Specifically, as long as overhearing a REP packet confirms an overlapped slot, the sender of the REP packet will remove the slot it intends to reserve on its time schedule time and try to reserve later. Therefore, we will overestimate the traffic rate of DATA packet if counting all time slots reserved by control packets in. To avoid this case, we approximate λ_{dc} as the sum of the traffic rate of DATA packets sent by this node, the traffic rate of DATA packets sent by all neighbors, and the traffic rate of non-overlapped DATA packets sent by hidden terminals. These three parts correspond to the three items in equation

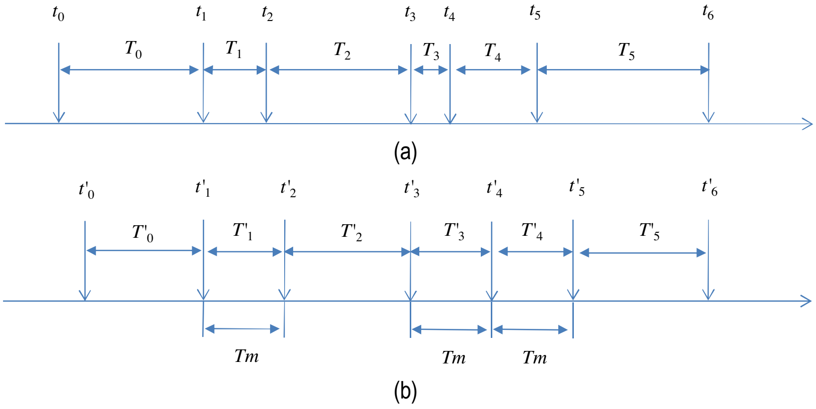

Figure 5. (a) Sequence of DATA packet arrival time and (b) Sequence of DATA packet sending time after being scheduled $(t_1^{'} = t_1,\ t_3^{'} = t_3,\ t_6^{'} = t_6)$.

(10), respectively. For the last item, we resort to a constant factor α, which can be trained with the simulation data, to obtain the traffic rate of non-overlapped DATA packets since it is complicated to model the probability of the overlapped packets. Therefore, λ_{dc} can be expressed as

$$\lambda_{\text{dc}} = \lambda_{\text{D}} + n\lambda_{\text{D}} + \alpha nk\lambda_{\text{D}} = (1 + n + \alpha nk)\lambda_{\text{D}}. \qquad (10)$$

In equation (10), since a node will send out a DATA packet only if the handshake is successful, λ_{D} can be evaluated by the following equation.

$$\lambda_{\text{D}} = \tilde{P_{\text{H}}}\lambda. \qquad (11)$$

Based on the distribution of T_i and the two conditions, we can calculate the probability $P\left(L = l|t_1 = t_1^{'}\right)$ of l successive intervals are all adjusted to intervals of length T_{m} as

$$\begin{aligned} P\left(L = l|t_1 = t_1^{'}\right) &= \int_0^{T_{\text{m}}} \int_0^{2T_{\text{m}}-T_1} \cdots \int_0^{lT_{\text{m}}-\sum\limits_{j=1}^{l-1} T_j} \\ &\times \int_{(l+1)T_{\text{m}}-\sum\limits_{j=1}^{l} T_j}^{\infty} \left(\lambda_{\text{dc}} e^{-\lambda_{\text{dc}} T_{\text{m}}}\right)^{l+1} dT_{l+1} dT_l \cdots dT_2 dT_1 \\ &= {\lambda_{\text{dc}}}^{l} e^{-(l+1)\lambda_{\text{dc}} T_{\text{m}}} \int_0^{T_{\text{m}}} \int_0^{2T_{\text{m}}-T_1} \cdots \int_0^{lT_{\text{m}}-\sum\limits_{j=1}^{l-1} T_j} \\ &\times dT_l \cdots dT_2 dT_1. \end{aligned} \qquad (12)$$

Note that equation (12) also denotes the probability that the $(l + 1)^{\text{th}}$ interval is long enough such that the arrival time of the next packet will not be adjusted. Thus, each T_{l+1} is actually one case of T_0. Then, we can get

$$P\left(t_1 = t_1^{'}\right) = \sum_{l=1}^{\infty} P_l. \qquad (13)$$

With $P\left(t_1 = t_1^{'}\right)$ and equation (12), we can calculate $\delta_l(T)$, the probability that an interval of length $T > T_{\text{m}}$ is reduced to the interval of length T_{m} among a given l successive adjusted interval.

$$\begin{aligned} \delta_l(T) = P\left(t_1 = t_1^{'}\right) \sum_{i=c}^{l} \lambda^l \text{e}^{-(l+1)\lambda T_{\text{m}}} \int_0^{L_{U_1}} \\ \cdots \int_0^{L_{U_j}} \cdots \int_{T_i=T}^{T+\epsilon} \cdots \int_0^{L_{U_b}} \cdots \int_0^{L_{U_l}} \\ dT_l \cdots dT_b \cdots dT_i \cdots dT_j \cdots dT_1, \end{aligned} \qquad (14)$$

where $\epsilon \rightarrow 0^{+}$ and

$$\begin{aligned} c &= \left\lceil \frac{T}{T_{\text{m}}} \right\rceil \\ L_{U_b} &= bT_{\text{m}} - T - \sum_{b=1}^{b-2} T_b \\ L_{U_j} &= \begin{cases} jT_{\text{m}} - \sum_{b=1}^{j-1} T_b & \text{if } c \leq i - j \\ iT_{\text{m}} - T - \sum_{b=1}^{j-1} T_b & \text{otherwise} \end{cases} \end{aligned} \quad . \qquad (15)$$

Furthermore, we can calculate the probability of interval of length $T(T > T_{\text{m}})$ is adjusted to T_{m} as

$$\delta(T) = P\left(t_1 = t_1^{'}\right) \sum\nolimits_{l=2}^{\infty} \delta_l(T). \qquad (16)$$

With $\delta(T)$[3], we can calculate the distribution of $T_{\text{I}}^{'}$ as follows

$$\begin{aligned} &P\left(T_{\text{I}}^{'} = T\right) \\ &= \begin{cases} \int_0^{T_{\text{m}}} \lambda e^{-\lambda T^{'}} dT^{'} + \int_{T_{\text{m}}}^{\infty} \delta\left(T^{'}\right) dT^{'} & \text{if } T = T_{\text{m}} \\ \lambda e^{-\lambda T} \epsilon - \delta(T) & \text{if } T > T_{\text{m}} \\ 0 & \text{otherwise} \end{cases} \cdot \end{aligned} \qquad (17)$$

Finally, with the distribution of T'_{I}, we can calculate $\tilde{P}_{\mathrm{H}}$ as

$$\tilde{P}_{\mathrm{H}} = \sum\nolimits_{T'_I} \left(P\left(\mathrm{H} | T'_I = T\right) P\left(T'_I = T\right)\right). \qquad (18)$$

In equation (18), since both sides contain $\tilde{P}_{\mathrm{H}}$, it is not easy to get a closed form of $\tilde{P}_{\mathrm{H}}$. Therefore, we resort to the numerical method (iteration) to calculate $\tilde{P}_{\mathrm{H}}$.

5. Performance evaluation

In this section, we use simulations to evaluate the performance of GOAL. Aqua-Sim [24], a NS-2 based underwater acoustic network simulator developed by the UWSN lab at the University of Connecticut, has been used for our simulations.

5.1. Simulation settings

In the simulation, nodes are randomly deployed within a 300 m × 300 m × 500 m area. When a node detects an event, it will send the data collected to the sink node. To simplify the simulations, we make two assumptions: (i) a node can detect the event occurring within its sensing range and (ii) event lasts for a long period of time[4], such that nodes send data to the sink node periodically as long as it can sense the event. This period is defined as sensing interval.

All nodes can move freely in horizontal two-dimensional space, i.e. in the *X*–*Y* plane. The speed of a node follows a uniform distribution between 0.2 and 1.5 m s^{-1}. The transmission range is set to 120 m. The sink node, which is the destination for all data packets, is fixed at (250, 250, 0). The sensing range of nodes is 80 m. The number of maximum retransmission times is set to be 6. Each simulation lasts for 5000 s. The energy consumption parameters are based on a commercial underwater acoustic modem, UMW 1000, from LinkQuest [25]: the power consumption on transmission mode is 2 W; the power consumption on receive mode is 0.75 W; and the power consumption on sleep mode is 8 mW.

Three metrics are used to quantify the performance: packet delivery ratio, energy consumption per byte, and delivery delay. Specifically, the packet delivery ratio is the ratio of the total number of packets sent by source nodes to the number of packets received by the sink node. The energy consumption per byte is to divide the total network energy consumption by the number of data bytes successfully received by the sink. The delivery delay is the average end-to-end delay of each packet received by the sink. We compare the performance of GOAL with VBF coupling with that of broadcast MAC (we use VBF for short in the rest of this work) [7].

5.2. Simulation results

Impacts of data sensing interval. In this set of simulations, the number of nodes in the network is fixed to be 100 and the size of the DATA packet is set to 300 bytes. Then, we change the data sensing interval of every node from 20 to 70 s.

As shown in Figures 6, 7, and 8, GOAL can provide a high end-to-end reliability. Figure 6 shows that GOAL can provide a much higher packet delivery ratio than VBF. This is because GOAL can greatly reduce collisions by its REQ/REP handshake process and its channel reservation mechanism. Additionally, we can see that the packet delivery ratio of GOAL increases while the sensing interval becomes larger. This is because nodes with a larger sensing interval generate fewer packets, which causes fewer collisions. Since the number of maximum retransmission times is fixed, the packet delivery ratio is improved when there are fewer collisions. We can also observe that the packet delivery ratio of VBF does not vary much while the sensing interval increases. This is because VBF is a best-effort protocol and the collision probability of VBF mainly depends on the self-adaptation scheme, which is highly related to the node distribution. Note that the size of network is fixed and nodes are uniformly deployed. Hence, the node distribution is decided by node density. In this simulation set, node density is fixed and therefore the packet delivery ratio keeps nearly the same value.

GOAL can also achieve high energy efficiency. From Figure 7, we can observe that GOAL is more energy-efficient than VBF, especially when the sensing interval becomes larger. This is because in GOAL, when the sensing interval is shorter, multiple packets can be sent together with just one REQ/REP handshake, which can improve the system's energy efficiency. In addition, as the sensing interval becomes larger, less data packets

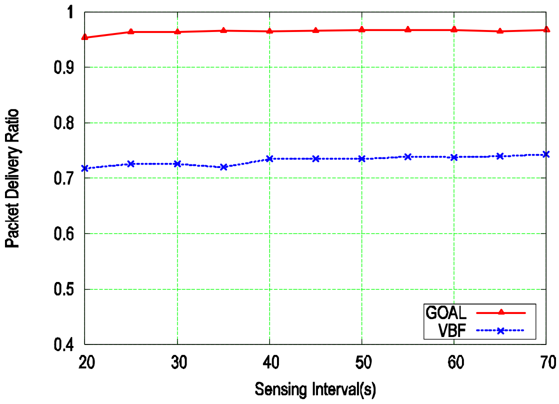

Figure 6. Packet delivery ratio with varying sensing interval.

[4] This is practical. For example, oceanic volcano usually belches slight smoke and ashes for a long time before it finally erupts.

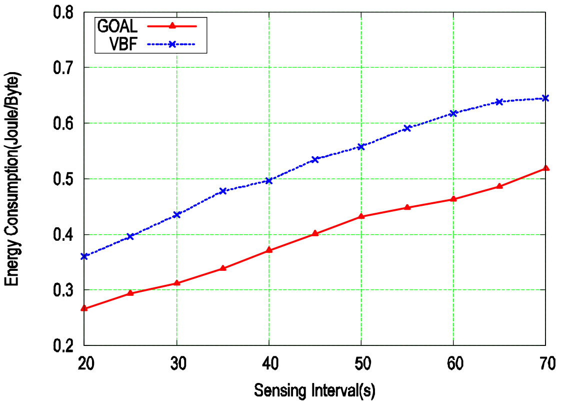

Figure 7. Energy consumption with varying sensing interval.

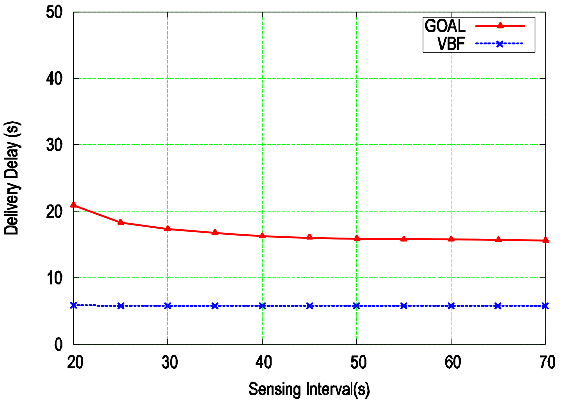

Figure 8. Delivery delay with varying sensing interval.

are sent in the network. Therefore, most nodes will waste energy in the idle state with a constant rate (8 mW). Additionally, this increases the energy consumption when the sensing interval is larger.

Considering the reliability requirement, the energy consumption in VBF is much higher than that in GOAL. For example, let us set P_{G} as the delivery ratio of GOAL and P_{V} as the delivery ratio of VBF, and set E_{G} and E_{V} as the energy consumption of GOAL and VBF, respectively. To achieve the same packet delivery ratio, VBF should perform retransmission for N times on average and thus the energy consumption is NE_{V}, where N satisfies

$$1-(1-P_{\mathrm{V}})^{N}=P_{\mathrm{G}}. \tag{19}$$

Hence, N can be expressed as

$$N=log_{1-P_V}(1-P_G). \tag{20}$$

In Figure 6, for example, the packet delivery ratio of GOAL and VBF is approximately 0.97 and 0.73, respectively, when the sensing interval is 50 s. Applying the above equation, we can obtain the average number of times that VBF should transmit each packet to reach the same packet delivery ratio as GOAL, which is as follows:

$$N=\log_{1-0.73}(1-0.93)=2.67. \tag{21}$$

Therefore, the energy consumption of VBF should be at least doubled. In other words, the energy consumption in GOAL is less than half of that in VBF, which indicates that GOAL is more energy-efficient.

Figure 8 shows us that the end-to-end delay of GOAL decreases with the increasing sensing interval. This is because collisions increase when the sensing interval is shorter. With collisions, nodes have to initiate a new REQ/REP/DATA round to do retransmission, which introduces extra delay. As the sensing interval becomes larger, fewer collisions and retransmission appear. Therefore, the delay decreases while the sensing interval increases. For VBF, which is a best-effort protocol, the delivery delay has almost nothing to do with traffic rate, but is mainly decided by the backoff time in the self-adaptation scheme. Thereby, the delivery delay in VBF does not change significantly in Figure 8.

Impacts of node density. In this set of simulations, we set the sensing interval of every node to be 50 s and change the number of nodes in the network from 70 to 120. The size of the DATA packet is fixed to be 300 bytes.

The impact of node density is shown in the next three figures. In Figure 9, we can see that the packet delivery ratio of GOAL is much higher than that of VBF. Again, this is because GOAL reduces more collision than VBF and VBF is a best-effort protocol. Also, we can see that the packet delivery ratio of both GOAL and VBF increases while there are more nodes in the network. One reason as mentioned before is that GOAL largely reduces the MAC collision by performing reservation for DATA packets. The other reason is that it is related to the self-adaptation scheme. Specifically, when the node density is lower, there are fewer qualified next-hops according to the self-adaptation scheme. Particularly, some forwarders do not have a qualified next-hop. In VBF, the DATA packet is dropped in such a case. In GOAL, forwarding failure can be detected by missing the implicit acknowledgement, and therefore a retransmission is issued.

From Figure 10, we can observe that GOAL consumes less energy than VBF for transmitting every unit data from source to sink. The reason is similar to that of Figure 7. In VBF, the collision probability is higher than that in GOAL. Moreover, each collided packet in VBF wastes more energy than that in GOAL because the packet in VBF is much longer. As a result, GOAL saves more energy. Similar to the analysis for Figure 7, if we analyze the energy consumption with the same packet

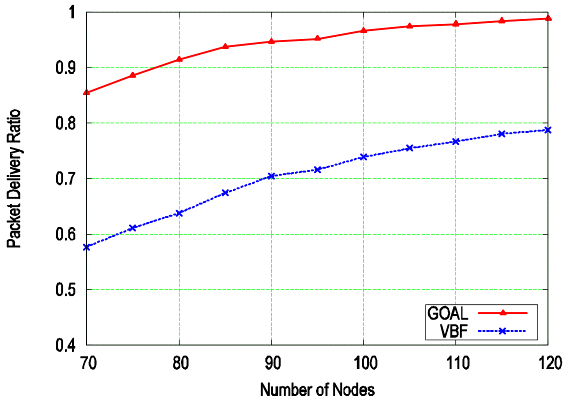

Figure 9. Packet delivery ratio with varying number of nodes.

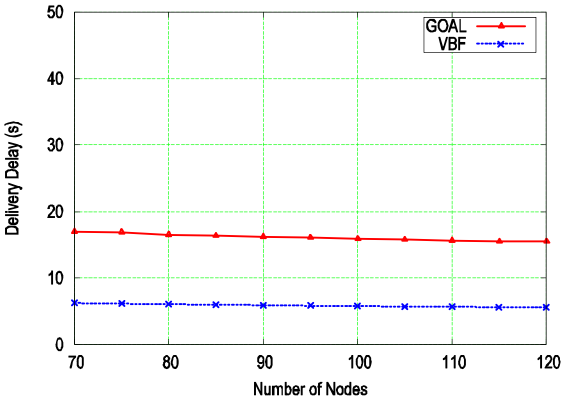

Figure 11. Delivery delay with varying number of nodes.

delivery ratio, we can see that the energy consumption in GOAL is much less than that in VBF, especially when there are less nodes within the network.

In Figure 11, the delivery delay of GOAL is higher than that of VBF. The reason has been mentioned before: the handshake and implicit acknowledgement in GOAL introduce more delay while VBF is a best-effort protocol which does not care whether the transmission to next-hop is successful. Due to the same reason, the delivery delay of VBF is almost a constant in Figure 11. Additionally, we can observe that the delivery delay of GOAL slightly decreases while the node deployment becomes dense. This is because dense deployment improves the probability that the next-hop with a better location can be found. According to the self-adaptation algorithm, a better location implies a shorter backoff time. Therefore, the total delivery delay is reduced.

Impacts of DATA packet size. In this set of simulations, we set the sensing interval and number of nodes as 50 s and 100, respectively. Then, we compare the performance of GOAL and VBF plus broadcast MAC by varying the size of DATA packet from 200 to 400 bytes with step 20.

From Figure 12, as the size of the DATA packet increases, the packet delivery ratio of VBF decreases. This is because a longer DATA packet increases the probability of collisions. However, as shown in Figure 12, the packet delivery ratio of GOAL is almost constant. We believe this is because the reservation scheme in GOAL well prevents the collisions among DATA packets.

From Figure 13, we can see that both GOAL and VBF consume less energy as the size of the DATA packet becomes larger. This is because the longer DATA packet shortens the time of the idle state. We can still see that GOAL is more energy-efficient than VBF. The reason is the same as mentioned before. The reservation scheme in GOAL can largely reduce the collisions among DATA packets and therefore GOAL wastes less energy.

From Figure 14, we can observe that GOAL introduces more delivery delay than VBF. The reason is same as the explanation for Figure 11. We can also observe that the delivery delays of both GOAL and VBF become longer while the DATA packet size increases slightly. This is reasonable. A longer DATA packet implies longer transmission delay. Since transmission delay is a part of delivery delay, the delivery delay becomes longer.

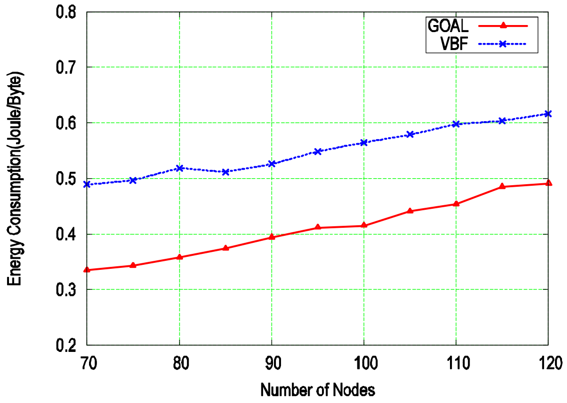

Figure 10. Energy consumption with varying number of nodes.

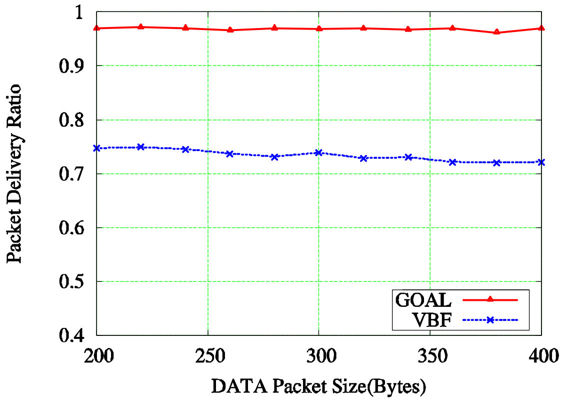

Figure 12. Packet delivery ratio with varying DATA packet size.

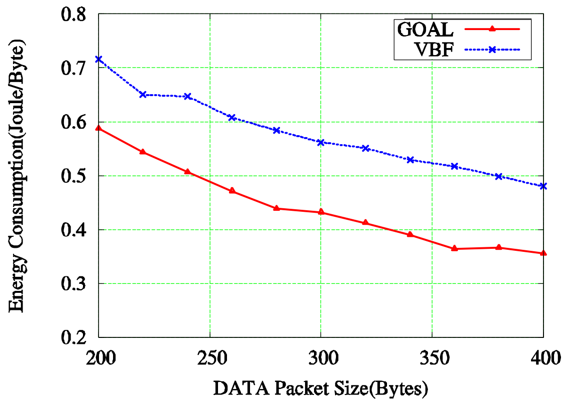

Figure 13. Energy consumption with varying DATA packet size.

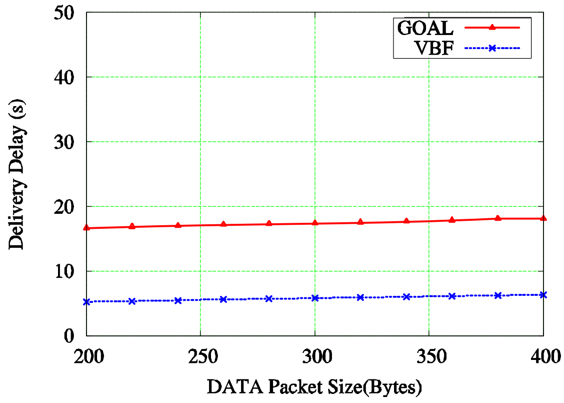

Figure 14. Delivery delay with varying DATA packet size.

5.3. Theoretical model verification

In this section, we perform two sets of simulation to verify the theoretical model of $\tilde{P}_{\mathrm{H}}$. In the first set, GOAL is revised as mentioned in Section 4 and the relay traffic is disabled. With the results of this set, we can check the model accuracy. After all, it is not a general scenario if nodes do not relay packet. Therefore, we let nodes still forward DATA packets in the other simulation set. By comparing with the results of this set, we can find the gap between our model and the simulation in practical scenario. There are 200 stationary nodes with a transmission range of 100 m uniformly distributed within the network of size 500 m × 500 m × 500 m, and each node randomly selects a destination node. Then, we vary λ from 0.01 to 0.1 packets s^{-1} with step 0.01 to evaluate $\tilde{P}_{\mathrm{H}}$.

The comparison between simulation results and our theoretical model is shown in Figure 15. From Figure 15, we can observe that the $\tilde{P}_{\mathrm{H}}$ evaluated by our theoretical model is always tightly around the simulation results of GOAL without forwarding. This justifies that our theoretical model matches the simulation results, which also indicate that our model can well describe the probability of successful handshake.

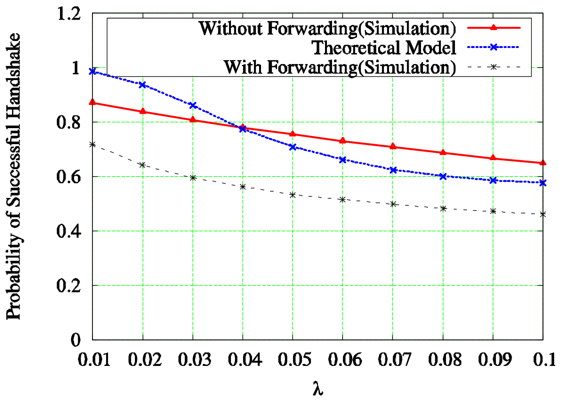

Figure 15. Comparison between theoretical model of $\tilde{P}_{\mathrm{H}}$ and simulation results.

At the same time, we can see the results of both the theoretical model of GOAL without forwarding are higher than that of GOAL with forwarding. This is reasonable. In GOAL with forwarding, nodes also send relay traffic to the channel and therefore the traffic in the channel is apparently much heavier than that in GOAL without forwarding. For that reason, there must be more collisions, which reduce the probability of successful handshake.

6. Conclusion

In this paper, GOAL, an efficient geo-routing aware MAC protocol, is proposed for underwater sensor networks. It is a reservation-based MAC protocol which can smoothly integrate with any existing geo-routing protocols with self-adaptation capability. Self-adaptation based REQ/REP handshake, geographic cyber carrier sensing, and implicit acknowledgement are used in GOAL to improve system performance. Although the end-to-end delivery delay increases because of the hop-by-hop retransmission mechanism in GOAL, it can achieve high end-to-end delivery ratio with low energy consumption. Plentiful simulation results show that GOAL outperforms existing VBF with broadcast MAC in both end-to-end delivery ratio and energy efficiency. Moreover, the simulation results demonstrate that our theoretical model can well describe the probability of a successful handshake.

References

[1] Cui, J., Kong, J., Gerla, M. and Zhou, S. (2006) Challenges: building scalable mobile underwater wireless sensor networks for aquatic applications. *IEEE Network, Spec. Issue Wireless Sens. Networking* **20**(3): 12–18.

[2] Heidemann, J., Ye, W., Wills, J., Syed, A. and Li, Y. (2006) Research challenges and applications for underwater sensor networking. In *Proceedings of IEEE Wireless*

Communications and Networking Conference (IEEE), 228–235.

[3] Partan, J., Kurose, J. and Levine, B.N. (2006) A survey of practical issues in underwater networks. In *Proceedings of ACM WUWNet* (ACM), 1–8.

[4] Liu, L., Zhou, S. and Cui, J. (2008) Prospects and problems of wireless communication for underwater sensor networks. *Wireless Commun. Mobile Comput.* **8**(8): 977–944.

[5] Chitre, M., Shahabudden, S. and Stojanovic, M. (2008) Underwater acoustic communication and networks: recent advances and future challenges. *Mar. Technol. Soc. J.* **42**(1): 103–116.

[6] Perkins, C.E. and Royer, E.M. (1999) Ad hoc on-demand distance vector routing. In *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications* (IEEE), 90–100.

[7] Xie, P., Cui, J. and Lao, L. (2006) VBF: vector-based forwarding protocol for underwater sensor networks. In *Proceedings of IFIP Networking* (Springer), 228–235.

[8] Xie, P., Zhou, Z., Peng, Z., Cui, J. and Shi, J. (2009) Void avoidance in three-dimensional mobile underwater sensor networks. In *Proceedings of International Conference on Wireless Algorithms, Systems, and Applications (WASA)* (Springer), 305–314.

[9] Yan, H., Shi, Z. and Cui, J. (2008) DBR: depth-based routing for underwater sensor networks. In *Proceedings of IFIP Networking* (Springer), 72–86.

[10] Cheng, X., Shu, H., Liang, Q. and Du, H. (2008) Silent positioning in underwater acoustic sensor networks. *IEEE Trans. Veh. Technol.* **57**(3): 1756–1766.

[11] Erol, M., Vierira, L.F.M. and Gerla, M. (2007) AUV-aided localization for underwater sensor networks. In *Proceedings of International Conference on Wireless Algorithms, Systems and Applications (WASA)* (Springer), 44–51.

[12] Frampton, K.D. (2006) Acoustic self-localization in a distributed sensor network. *IEEE Sens. J.* **6**(1): 166–172.

[13] Zhou, Z., Cui, J. and Zhou, S. (2007). Localization for large scale underwater sensor networks. In *Proceedings of IFIP Networking* (Springer), 108–119.

[14] Zhou, Z., Cui, J. and Bagtzoglou, A. (2008) Scalable localization scheme with mobility prediction for underwater sensor networks. In *Proceedings of IEEE INFOCOM* (IEEE), 2198–2206.

[15] Xie, P., Zhou, Z., Cui, J. and Shi, Z. (2007) R-MAC: an energy-efficient MAC protocol for underwater sensor networks. In *Proceedings of International Conference on Wireless Algorithms, Systems and Applications (WASA)* (Springer), 187–198.

[16] Park, M.K. and Rodoplu, V. (2007) UWAN-MAC: an energy-efficient MAC protocol for underwater acoustic wireless sensor networks. *IEEE J. Oceanic Eng.* **32**(3): 710–720.

[17] Syed, A., Ye, W. and Heidemann, J. (2008) T-Lohi: a new class of MAC protocols for underwater acoustic sensor networks. In *Proceedings of IEEE INFOCOM* (IEEE), 231–235.

[18] Fullmer, C.L. and Garcia-Luna-Aceves, J. (1995) Floor acquisition multiple access (FAMA) for packet-radio networks. In *Proceedings of ACM SIGCOMM* (ACM), 262–273.

[19] Molins, M. and Stojanovic, M. (2006) Slotted FAMA: a MAC protocol for underwater acoustic networks. In *Proceedings of IEEE OCEANS* (IEEE), 16–19.

[20] Peng, Z., Zhu, Y., Zhou, Z. and Cui, J. (2010) COPE-MAC: a contention-based medium access control protocol with parallel reservation for underwater acoustic networks. In *Proceedings of IEEE OCEANS* (IEEE), 1–10.

[21] Zhou, Z., Peng, Z., Cui, J. and Shi, Z. (2009) Analyzing multi-channel MAC protocols for underwater acoustic sensor networks. *UCONN CSE Technical Report: UbiNet-TR08-02.*

[22] Xiao, Y., Zhang, Y., Gibson, J.H. and Xie, G.G. (2009) Performance analysis of p-persistent aloha for multi-hop underwater acoustic sensor networks. In *Proceedings of ICESS* (IEEE), 305–311.

[23] Bianchi, G. (2000) Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE J. Sel. Areas Commun.* **18**(3): 535–547.

[24] Xie, P., Zhou, Z., Peng, Z., Yan, H., Hu, T., Cui, J., Shi, Z. *et al.* (2009) Aqua-Sim: an NS-2 based simulator for underwater sensor networks. In *Proceedings of IEEE/MTS Oceans* (IEEE), 1–7.

[25] LinkQuest, http://www.link-quest.com.