DSH-MAC: Medium Access Control Based on Decoupled and Suppressed Handshaking for Long-delay Underwater Acoustic Sensor Networks

Tiansi Hu and Yunsi Fei
Department of Electrical and Computer Engineering
Northeastern University, Boston, Massachusetts 02115
Email: {tiansi, yfei}@ece.neu.edu

***Abstract*—Efficient underwater networking is still a challenging issue due to its physical limitations, like long propagation delay. In this paper, we focus on medium access control (MAC) for underwater acoustic sensor networks (UW-ASNs). Considering that the handshaking process in traditional contention-based MACs is the main hurdle for improving the network channel utilization, we propose a novel MAC protocol with Decoupled and Suppressed Handshaking (DSH-MAC) in order to reduce the time overhead, and therefore achieve more efficient channel utilization. In DSH-MAC the conventional two-way handshaking is decoupled, and hence relevant nodes are able to perform other transmissions while control packets are propagating in water. DSH-MAC also suppresses unnecessary control packets with traffic prediction, further improving the channel utilization and throughput. Our proposed protocol has been proven to be channel-efficient with both theoretical analysis and intensive simulations.**

I. Introduction

Underwater Acoustic Sensor Networks (UW-ASN) [1], which will enable a variety of aquatic applications, has been actively investigated over the past decade. However, efficient underwater networking is still a challenging open problem due to the adverse underwater environment.

As radio signals do not propagate well in water, underwater communications feature acoustic signals, bringing distinctive properties and challenges in underwater communications and networking. The propagation delay is 5-order longer than RF signals, and therefore is dominant in the total communication time. For example, considering transmitting a 64-byte packet to a node 500 meters away, the propagation delay is approximately 333 ms while the transmission time is 50 ms (at 10 kbps). Therefore, better utilizing the propagation time will significantly help improve the network throughput, especially in UW-ASNs where the data transmission rate is low.

Medium Access Control (MAC) layer sits right above the physical layer (PHY) and manages the shared communication medium, by coordinating the access times of a number of nodes. Therefore, it has great impact on network performances, including delay, throughput, fairness, and energy consumption. Underwater MAC has to be designed to suit the physical media properties.

Our goal is to design a high-throughput and energy-efficient underwater MAC protocol adaptive to the dynamic underwater environment. We focus on making the coordination and handshaking process between nodes more efficient.

In the underwater environment, the network topology can be changed by several causes, including node mobility with water currents, link disruptions due to poor acoustic channels, and failures of node hardware. Contention-based random access MAC protocols can react to network dynamicness well, as they do not need to maintain much neighbor information and links are established on-demand. However, adopting contention-based MAC protocols in UW-ASNs will result in poor performance. First, the two-way handshaking before data transmission in contention-based MAC protocols introduces large propagation delay overhead. When collisions happen during the handshaking process, the delay will be even longer and more energy is required for retransmitting. Second, during the handshaking process, the involved nodes (neighbor nodes of the sender and/or the receiver) cannot transmit, resulting in a large waste of communication channel and consequently low network throughput.

We observed that certain rules in the traditional RTS/CTS handshaking process do not work any more in UW-ASNs, where the propagation delay is non-negligible and dominates in the communication time. The traditional RTS/CTS-based MAC reserves the communication channel for a pair of sender and receiver for both the handshaking process and data-communication process. This is over-conservative. For example, after sender A sends RTS to B, a neighbor of A hidden to the receiver B should be allowed to send its data request as well, because it is not in the transmission range of B and will not cause any collisions at B. As collisions only happen at the receiver, we propose to redesign underwater MAC to be receiver-initiated. The receiver arbitrates among its neighbors and invites the most appropriate one to send data. The arbitration is made based on some information about its neighbors (senders) the receiver collects before. In this way, as the receiver controls the data communication, collisions on the receiver will be greatly reduced. The receiver still relies on some control packets (similar to RTS) sent by its neighbors to learn the sending intention, but these control packets are not sent back to back with invitation. They can be piggybacked with data transmissions and one round of propagation delay will be saved. Therefore, the overhead is reduced significantly.

978-1-4799-0537-9/13/$31.00 ©2013 IEEE

In this paper, we propose a MAC protocol, DSH-MAC, designed for UW-ASNs to address their long propagation delay issue, based on active selection of sender by the receiver in an intelligent way. In DSH-MAC, The two control packets, namely NOTE and GRANT, are analogous to RTS and CTS packets, respectively. The NOTE packets are used to notify others of the sender's transmission intention, including the number of packets buffered for each receiver and the data generation rate. The GRANT packets are used to inform the sender the readiness of the channel. The two control packets are decoupled so that parallelism is allowed between handshaking/handshaking and handshaking/data transmissions, leading to higher network throughput.

One main difference between DSH-MAC and the previous work is that the NOTE packets are only for notification of the sender's intent and they do not keep the other neighbors of the sender silent. GRANT packets are not responding to NOTE packets, and the sender does not expect any feedback after sending out NOTE. Therefore the sender is able to perform other operations after sending NOTE, e.g., sending GRANT to other nodes. GRANT packets are similar to CTS packets in terms of granting the channel and suppressing other neighbors of the receiver from sending anything in the upcoming data transmission. However, they are decoupled from NOTE packets, and NOTE packet do not precede every data transmission any more. Multiple control packets can be sent in parallel and control packets and data packets can be send in parallel, greatly increasing the network throughput.

The rest of the paper is organized as follows. In Section II, we review the related work. In Section III we present DSH-MAC in detail. We analyze the performance of DSH-MAC in Section IV, and demonstrate our simulation results in Section V, followed by conclusions in Section VI.

II. Related Work

We next review the related work on MAC protocols designed for underwater sensor networks.

There has been a lot of MAC protocols proposed for both terrestrial and underwater sensor networks. In terms of the way that nodes coordinate, they can be divided into mainly two categories: *contention-free* and *contention-based* MAC protocols. Although the *contention-based* protocols are able to handle dynamic networks well as they do not depend on fixed resource allocation among nodes, they suffer from the long underwater propagation delay. For *contention-free* protocols [2–5], once consent on channel allocation is achieved, the nodes take their turns in using the channel. However, they either lack adaptiveness to underwater topology changes or have low and unfair utilization of the common resources. Moreover, to ensure a collision-free schedule, the cycle time is set very long [3], resulting in low throughput and high delay.

There are several contention-based MAC protocols designed for UW-ASNs, such as Slotted-FAMA [6], PCAP [7], and TLohi [8]. Slotted-FAMA and PCAP aim at reducing collision rate incurred by long propagation delay. However, they sacrifice the channel utilization (throughput). TLohi [8] is a tone-based MAC protocol for energy efficiency. It requires extra hardware - a tone receiver in the acoustic modem, and significant delay still exists in the contention process (handshaking).

There are also some contention-based MAC protocols designed to improve the handshaking efficiency, especially with the long propagation delay in underwater environment, such as COPE-MAC [9], CS-MAC [10], PDAP [11], and Grant-to-send [12]. However, COPE-MAC only works in static networks, where the propagation delay between neighbors is assumed constant. CS-MAC provides more transmission chances during the handshaking process by allowing some parallelism during its handshaking phase, but the gain of the network throughput is limited in a sparse underwater sensor network. PDAP includes propagation delay information in control packets, so that some interferences from the overhearing nodes can be avoided. However, PDAP does not change the fundamental RTS/CTS handshaking, the improvement on the control overhead is limited. Grant-to-send improves carrier sense multiple access (CSMA) protocols by allowing the receiver to initiate transmissions. However, the initiation lacks flexibility in choosing senders and thus its performance heavily depends on traffic patterns. Some common traffic patterns may lead to performance degradation, including throughput and fairness. At the same time, because it lacks handshaking, data packet collisions induce higher time and energy penalty.

III. The Protocol Design

In this section, we first give an overview of the protocol, DSH-MAC, followed by detailed algorithms.

A. Overview

There are two major considerations for a MAC protocol - throughput and fairness. The throughput is greatly affected by how nodes do handshaking. To improve the fairness, a MAC protocol should ensure that the channel is shared properly among the nodes/packets. The fairness is determined by how receivers prioritize their senders and choose among them.

We propose a new way of handshaking tailored for long-delay communication channels in UW-ASNs. In conventional contention-based MAC protocols, during the RTS/CTS handshaking process which is approximately twice the single propagation delay, all the neighboring nodes have to keep silent, resulting in low channel utilization and throughput.

As a random access MAC protocol, DSH-MAC is changing the way that handshaking is carried out. In Fig. 1, we compare the process of data transmissions in DSH-MAC (Fig. 1b) to a typical RTS/CTS MAC protocol (Fig. 1a), and we can observe two advantages of our protocol:

- **Shorter handshaking with receiver-initiated communication.** In conventional RTS/CTS-based MAC (Fig. 1a), a round of data communication involves transmission and propagation of RTS, CTS and DATA. While in DSH-MAC, as in Fig. 1b, one round of communication only involves GRANT and DATA. NOTE packet is piggy-backed with data, and its propagation delay is eliminated.

Its purpose is to let the receiver know the current buffer status so that the receiver is able to prepare later data transmission.

- **Efficient channel utilization with parallel collision-free communications.** Considering the sender and its neighbors who are hidden from the receiver (i.e., not in the transmission range of the receiver), the sender does not need to wait after sending a NOTE packet, nor its neighbors upon overhearing the NOTE. They both are free to carry out their own transmissions rather than keeping silent while the receiver is receiving data and NOTE packets. This allows more parallelism between control packets or control packet and data packet, and eliminates unnecessary waiting. The overall data throughput is further improved.

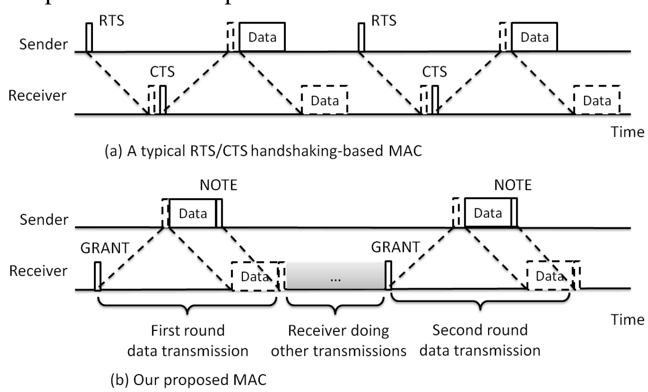

Fig. 1. A comparison of the data transmission process

To achieve fairness, a receiver should not choose a sender arbitrarily. Instead, it should choose the sender that has the highest urgency, e.g., with the most packets and the highest queuing delay. We introduce a metric named queue index $\mathrm{QI}(A, B)$ that evaluates the urgency of outstanding transmissions from node A to node B. QI indicates both the length of the queue and the age of the packets, and it varies along data transmissions. Dissipating the packets in a queue, i.e., transmitting them, will lower the QI and therefore give chances to other neighbors. Neighbors with few packets and low generation rates will eventually obtain the channel grant instead of being starved, because the packet ages increase as the time elapses.

B. The protocol

We next present our proposed MAC protocol, DSH-MAC, in detail. Fig. 2 depicts components in the MAC protocol and the relations between two sensor nodes. Each node can act as a sender and receiver for different data traffic. Each node, as the sender, maintains a set of queues for its neighbors, each of which stores the outgoing packets to a neighbor. A node is responsible for notifying its neighbors of its queue status (in the form of QI) by sending NOTE packets so that the neighbors can arbitrate to call the node at an appropriate time.

Next, we explain the protocol and algorithms running on senders and receivers, and their interactions based on the status of queues (QI's).

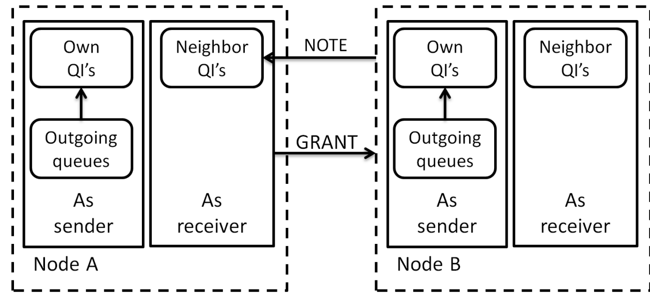

Fig. 2. Components in the MAC protocol

1) The sender: The MAC protocol on the sender addresses the issue of when to notify its neighbors (potential receivers) of its queue's status so that each receiver can schedule when to start a data transmission between them.

A NOTE packet from sender node A to receiver node B (denoted as NOTE(A,B)) contains the following values:

- $\mathrm{QI}(A, B)$: the queue index. It is the sum of queuing time of all the buffered packets for node B. The higher the QI, the more urgent to send.
- N and ΔN: number of packets for node B in A's outgoing queue, and the average packet generation rate for node B, respectively. They are used for node B to extrapolate node A's time-varying $\mathrm{QI}(A, B)$ during the time that no NOTE transmission takes place between node A and B. Both values are non-negative.

A NOTE packet is sent by node A in two cases: 1) a NOTE packet is always piggybacked with a data packet or merged into its packet header, so as to inform neighbors of the sender's queue status without extra propagation delay; 2) the sender A realizes that the receiver B has an inaccurate estimation of $\mathrm{QI}(A, B)$.

The first case is straight-forward shown in Fig. 1. For the second case, there are two indicators that node B has an inaccurate estimation of $\mathrm{QI}(A, B)$ - absence of GRANT packets from B when the channel is idle or GRANT is given to a lower-priority neighbor of B.

Absence of GRANT packets means that while the channel is idle, node A does not hear or overhear any GRANT packets sent from node B to either itself or others. For example, when node A's traffic rate rises from 0, but node B is not aware of the traffic change based on the old extrapolation $\hat{\mathrm{QI}}(A, B)$, a NOTE packet needs to be sent by A to refresh node B's neighbor QI associated with A.

The overhearing of B's GRANT packet to its another neighbor with lower QI than A's QI is another case triggering A's QI update. As the GRANT packet contains the selected sender C and the estimated $\hat{\mathrm{QI}}(C, B)$. A can compare $\mathrm{QI}(A, B)$ with $\hat{\mathrm{QI}}(C, B)$ to decide whether an update NOTE packet from A is necessary.

Note that after sending a NOTE packet, the sender does not need to wait for any response from the receiver before doing other jobs, and the NOTE packets sent along with data do not contribute any propagation delay to data transmissions. As both GRANT and NOTE packets are very small, the chances of collision with other packets are negligible.

2) The receiver: In DSH-MAC, the receiver node needs to initiate data transmissions from its neighbors intelligently, based on the QI's received and its own estimations. Every node in the network should periodically estimate whether their neighbors have data packets to send.

As shown in Fig. 1b, there may be a gap between the time the receiver receives a NOTE packet from the sender and the time it sends a GRANT packet to the same sender (grey period in the figure). Therefore, the receiver has to project A's queue change in order to make a good decision on choosing the most urgent node.

As the total queuing time changes over time, the receiver should estimate the current QI value between it and the associated neighbor with first-order extrapolation:

$$\hat{\mathrm{QI}} = \mathrm{QI} + \Delta\, tN + \frac{1}{2}\Delta t^2 \Delta N, \quad (1)$$

where QI is the one contained in the NOTE packet and Δt is the time between the NOTE departure time and the current time. The second and the third terms are the increased amount of the queuing time contributed by the original packets and the new generated packets during that time, respectively. As Δt and ΔN represents the time elapsed after receiving the last QI and packet generation rate at the sender respectively, which are both non-negative, the estimated $\hat{\mathrm{QI}}$ is always larger or equal to zero.

When the channel is idle, the node checks whether there are any neighbors with packets to send according to its own estimation of $\hat{\mathrm{QI}}$'s. If then, it picks the neighbor $N_{\max}$ with the highest $\hat{\mathrm{QI}}$ and grants the channel to node $N_{\max}$ by sending a GRANT packet. The GRANT packet contains the current node ID (receiver), the ID of the selected sender, and the associated $\hat{\mathrm{QI}}$ between them, which is used by other overhearing neighbors of the receiver to compare with their own QI with and trigger update NOTE if estimation inaccuracy is detected. The receiver does not send GRANT packets to a node with $\hat{\mathrm{QI}} < \mathrm{QI}_{\min}$, which is set to 1.0. The transmission between the sender and receiver can be restarted either after $\hat{\mathrm{QI}}$ goes up above $\mathrm{QI}_{\min}$ or the sender sends a update NOTE to the receiver as in the case of absence of GRANT packets explained in III-B1.

IV. Analysis

In this section, we analyze the channel utilization of our protocol, and compare it to Slotted-FAMA [6]. For analysis and comparison, we assume that DSH-MAC is slotted, i.e., GRANT and DATA packets are allowed to be transmitted at the beginning of synchronized slots only. The slot length should accommodate the transmission time of a GRANT packet plus the maximum propagation delay between nodes in the network. NOTE packets can be sent at the end of DATA and when it is necessary, i.e., not necessarily at the beginning of slots, because NOTE transmissions are asynchronous and do not require any response due to the nature of our proposed protocol.

We assume a multi-hop network, where each node has N neighbors and each of the neighbors has H hidden neighbors (cannot be reached by the original node). In the example network shown in Fig. 3, $N = 6$ for node 0, and $H = 3$ (shaded nodes) for node 0's neighbors. Fig. 3 is just for illustration and does not show the entire network.

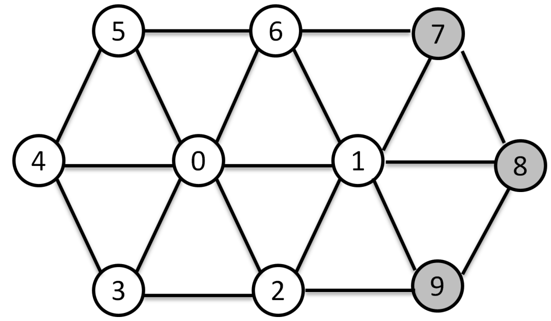

Fig. 3. An example of a multi-hop network

We let every node in the network generate packets at a rate of λ packets per second, following Poisson process, and the traffic is evenly distributed among its N neighbors, i.e., packets are generated to each of its neighbors at the rate of λ/N. We assume that the traffic rate is below the channel capacity, and therefore, in DSH-MAC, the rate of GRANT packets received by each node from all its neighbors is approximately λ. Note that this traffic setting simulates a network with multiple traffic flows, and with a given network-wide overall packet rate, the uniform packet rate creates highest packet collision rate compared to any non-uniform settings. In particular, a network with only one pair of nodes transmitting has no collisions.

The node channel utilization is defined as

$$U = \frac{\bar{T}_{\mathrm{user}}}{\bar{T}_{\mathrm{idle}} + \bar{T}_{\mathrm{busy}}}, \quad (2)$$

where $\bar{T}_{\mathrm{user}}$ is the average time for transmitting useful data excluding its propagation time, $\bar{T}_{\mathrm{idle}}$ is the average time that the channel is idle, and $\bar{T}_{\mathrm{busy}}$ is the average time that the channel is busy. Let the length of a slot be T_{slot}. The GRANT packets consume one slot each. Because NOTE packets are very small and the sender nodes do not need to wait after sending NOTE packets, the channel can be busy again for other operations while the NOTE packets are propagating in water. Therefore, we ignore the time used for NOTE packets.

Denoting the transmission time for a data packet as T_{tx} and the maximum propagation delay as T_{prop}, the total time consumed by a data packet is $T_{\mathrm{dslot}} = nT_{\mathrm{slot}} = \lceil T_{\mathrm{tx}} + T_{\mathrm{prop}} \rceil$, where $\lceil\cdot\rceil$ means rounding up to the next integral of slot.

With the network traffic following Poisson process, the average idle time between two successive transmissions for node 0 with its N neighbors is

$$\bar{T}_{\mathrm{idle}} = \frac{1}{(N+1)\lambda} \quad (3)$$

The average busy time $\bar{T}_{\text{busy}}$ includes two parts: the time spent due to successfully receiving (including overhearing) a GRANT packet and due to failure to receive (including overhearing) a GRANT packet. In order to derive the times in the two categories above, we calculate the probability of collisions first.

A. Probability of receiving or overhearing a GRANT packet

As a data transmission is performed after a node receives a GRANT packet, we now derive the probability that one given node, say node 0, is able to successfully receive a GRANT packet.

Let event A = {node 0 successfully receives or overhears a GRANT packet} and event B = {one of node 0's neighbors sends a GRANT packet}. Under the condition of event B, collisions may happen in two scenarios: 1) node 0 itself sends a GRANT packets in the same slot, and therefore, due to its transducer is in the transmission mode, it is not able to receive the GRANT packet from its neighbor; 2) one or more other neighbors send GRANT packets in the same slot, which causes collisions at node 0. No collision implies that none of the N nodes sends a GRANT packet, including node 0 and the rest $N-1$ neighbors other than the one in event B. Because each node broadcasts GRANT packet at an overall rate of λ, the probability that none of the N nodes sends a GRANT packet is $P\{A|B\} = \frac{(\lambda N T_{\text{slot}})^k e^{-\lambda N T_{\text{slot}}}}{k!}\Big|_{k=0} = e^{-\lambda N T_{\text{slot}}}$.

With the fact that every node transmits at the same rate, i.e., $P\{B\} = \frac{1}{N+1}$, the probability of node 0 successfully receives or overhears a GRANT packet is $P\{AB\} = \frac{1}{N+1}e^{-\lambda N T_{\text{slot}}}$ and the probability of node 0 fails to receive a GRANT packet is $P\{\bar{A}B\} = \frac{1}{N+1}(1-e^{-\lambda N T_{\text{slot}}})$. As the traffic rate and the number of neighbors are the same for every node, the above probabilities apply to any other nodes as well.

Next, taking node 0 as an example, we derive the busy time on success and failure to receive or overhear a GRANT packet.

B. Busy time on successfully receiving or overhearing a GRANT packet

When node 0 successfully receives or overhears a GRANT packet from one of the N neighbors, say node 1, it has to spend one time slot for a GRANT packet (T_{slot}) plus the time slots for a DATA packet (T_{dslot}) for either sending DATA or waiting for node 1 to receive DATA from others. With the probability of successfully receiving or overhearing a GRANT packet being $P\{A|B\}$, and the there are N neighbors, the average time used is

$$\bar{T}_{\text{rg}} = N(T_{\text{dslot}} + T_{\text{slot}}) \cdot P\{AB\} \tag{4}$$

in which, the time used to transmit user data from node 0 is

$$\bar{T}_{\text{user}} = N \cdot \frac{1}{N} \cdot T_{\text{tx}} \cdot P\{AB\} = T_{\text{tx}} \cdot P\{AB\} \tag{5}$$

In addition, node 0 needs to spend the same time ($T_{\text{dslot}} + T_{\text{slot}}$) after sending a GRANT to one of its neighbors and the neighbor successfully receives it, which means node 0 will receive data. As the GRANT receiving success rate is $P\{AB\}$ for each neighbor, and each of the N neighbors receives $1/N$th of node 0's GRANT packet, the average time used for node 0 receiving data is:

$$\bar{T}_{\text{sg}} = (T_{\text{dslot}} + T_{\text{slot}}) \cdot P\{AB\} \tag{6}$$

Besides, without receiving a GRANT packet, node 0 should keep silent due to carrier sensing when one of its neighbors successfully receives a GRANT packet from its hidden neighbors to node 0 and starts to send. For example, node 1 receives a GRANT packet from node 7 and it starts to transmit data in the next slot. As every neighbor has H hidden neighbors that node 0 cannot hear, the average amount of time that node 0 needs to keep silent is

$$\bar{T}_{\text{hidden}} = \frac{H}{N} \cdot N T_{\text{dslot}} \cdot P\{AB\} = H T_{\text{dslot}} \cdot P\{AB\} \tag{7}$$

C. Busy time on failure to receive a GRANT packet

There are two types of time spent in this category. First, when node 0 sends a GRANT packet to one of its neighbors, but the neighbor fails to receive. As a result, the transmission does not start. The total time spent is made up of a slot used to send the GRANT packet and a slot used to wait for the data packet and detect the error. Therefore, the average time spent for this case is

$$\bar{T}_{\text{fail}} = N \cdot \frac{1}{N} \cdot 2T_{\text{slot}} \cdot P\{\bar{A}B\} = 2T_{\text{slot}} \cdot P\{\bar{A}B\} \tag{8}$$

Second, one of node 0's neighbors may fail to overhear a GRANT packet and later start to send control or data packets when it should keep silent. As a result, node 0 hears noise, and in the worst case, node 0 has to wait for the whole data transmission period, which is T_{dslot}. As there are N neighbors, the average time is

$$\bar{T}_{\text{coll}} = N T_{\text{dslot}} \cdot P\{\bar{A}B\} \tag{9}$$

D. Channel utilization and discussion

After plugging everything in (3)-(9) to (2), we have $U = \bar{T}_{\text{user}}/(\bar{T}_{\text{idle}} + \bar{T}_{\text{rg}} + \bar{T}_{\text{sg}} + \bar{T}_{\text{hidden}} + \bar{T}_{\text{fail}} + \bar{T}_{\text{coll}})$, which is the channel utilization of a single node. In Fig. 4, we plot the it together with that of Slotted-FAMA presented in [6] with the optional ACK packet removed. The slot size is set to 0.4 s and the data transmission time is set to 0.3 s, which occupies two slots together with propagation delay. The channel utilization of both protocols increases as the channel demand increases until the saturation point ($\lambda = 0.12$ for Slotted-FAMA and $\lambda = 0.24$ for DSH-MAC), after which both decrease as traffic rate λ becomes larger due to higher handshaking failure rate. However, in the case of DSH-MAC, the channel saturates at a higher traffic rate than Slotted-FAMA, and the channel utilization decreases more slowly, which is because of two reasons. First, DSH-MAC has higher handshaking success probability compared to Slotted-FAMA. With higher handshaking success probability, less time is wasted due to additional attempts to retransmit. Besides, the time needed for a data packet is one slot less than Slotted-FAMA. Both features of DSH-MAC help achieving higher channel utilization, which has been confirmed by our simulations to be presented in Section V.

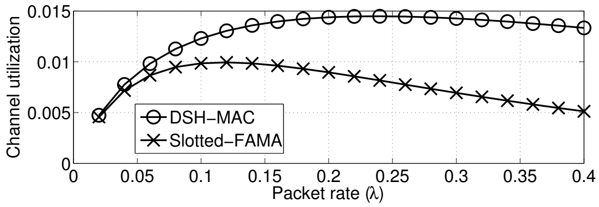

Fig. 4. Channel utilizations based on analysis

V. Performance Evaluation

In this section, we evaluate our proposed protocol, DSH-MAC by simulating it on a network simulator OMNet++-4.2.2 [13] with INET-1.99.4 [14] extension. We incorporate the underwater PHY, including the propagation delay, to the simulator. We first present our simulation settings, and then evaluate several network performance metrics such as packet average delay, control overhead, and fairness, in both a static network and a mobile network.

A. Simulation Settings

We deploy 20 nodes in a $1500\text{m} \times 2500\text{m}$ underwater area with a topology similar to the one shown in Fig. 3. The average neighbor distance is around 500 m. The maximum transmission range is set to 550 m. The channel bandwidth is set to 10 kbps. The other settings are the same as in Section IV. All experiments are run 30 times and the mean values and the 95% confidence intervals are shown in the result figures.

We compare DSH-MAC to three existing popular MAC protocols: Slotted-FAMA [6], UWAN-MAC [3], and Grant-to-send [12]. All the protocols are compared in terms of the following metrics:

- **Average delay:** this is the average time between packets are released to MAC and they are received by their recipients. The average delay is heavily impacted by both protocol throughput and the time used by handshaking. Specifically, when the traffic rate is lower than protocol throughput, the average delay is dominated by the time used by handshaking. Otherwise, the average delay is determined by protocol throughput.
- **Control overhead:** this is defined as the total number of control packets, which is an indicator of energy efficiency.
- **Fairness:** we evaluate two types of fairness - node fairness and packet fairness. There are many definitions of fairness metrics, and we adopt the one in [15]. The node fairness is defined as $F_{\text{node}} = \frac{(\sum_{i=1}^{N} \text{Throughput}_i)^2}{N\sum_{i=1}^{N} \text{Throughput}_i^2}$, where Thoughput_i is the throughput of the i-th node and N is the total number of nodes. We define the packet fairness as $F_{\text{packet}} = \frac{(\sum_{j=1}^{M} \text{Delay}_j)^2}{M\sum_{j=1}^{M} \text{Delay}_j^2}$, where Delay_j is the delay of the j-th packet and M is the total number of packets. Note that with the above definition, the fairness indices F_{node} and F_{packet} are in the range of $[\frac{1}{N}, 1]$ and $[\frac{1}{M}, 1]$, the higher values the more fair. The node fairness indicates whether each node has equal opportunity to transmit, whereas the packet fairness discloses whether packets are sent by nodes with the equal delay.

B. Static Networks

We first evaluate the prediction error of the QI on the receiver side. Fig. 5 depicts the prediction error rate on a node in the center of the network. We define the prediction error rate as the ratio between the error and the real QI value. We observe that when the traffic rate is low, the prediction error rate is around 50% which is high and decreases to around 5% when the traffic rate increases. This is because with low packet rates, NOTE packets are much less frequent. As a result, the receiver may not always have the most up-to-date information of the sender, resulting in a high prediction error.

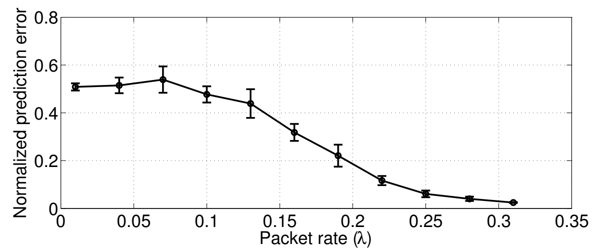

Fig. 5. Prediction error

In spite of the high prediction error rate at the low traffic rates, it does not affect the performance of the protocol. This is because at low traffic rates, the real QI is very low. Even with high relative estimated error, the estimated QI is very low as well, which does not exceed the threshold for the receiver to send a GRANT packet.

Fig. 6 depicts the average packet delay in low traffic (less than 0.07 packets per packet). In the figure, DSH-MAC has similar average packet delay compared to UWAN-MAC and Slotted-FAMA, but a little higher than Grant-to-send. As the packet rate increases, the delay of DSH-MAC grows slower than any other protocols. The reason is that with low traffic rates, the channel is not fully utilized due to the lack of packets demands rather than low efficiency of the protocol. Due to the CSMA nature of Grant-to-send, where there is no control packets, it achieves lowest delay. Because packets are able to be sent almost right away, the local packet queue is empty. Sending more GRANT packets than needed will not help increasing the number of data transmissions, but waste energy and channel resources.

With high traffic rates, DSH-MAC has large advantages over other types of MAC protocols, as shown in Fig. 7. When the traffic rate is high, packets have lowest queuing delay with DSH-MAC, which means that DSH-MAC has the highest data throughput. Furthermore, as the traffic rate increases, the average packet delays for all four protocols start to grow rapidly at certain traffic rates, which indicates that the corresponding protocol has reached its throughput saturation point. Among the four protocols, DSH-MAC has the highest saturation traffic rate ($\lambda = 0.22$), and Slotted-FAMA has the lowest ($\lambda = 0.1$), which is consistent with our

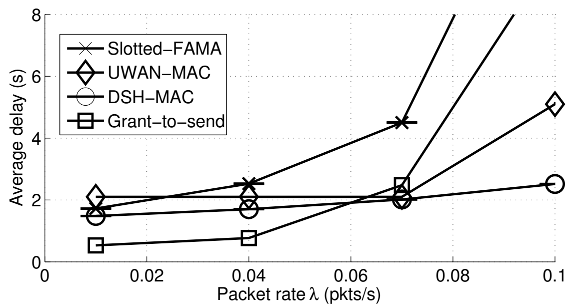

Fig. 6. Average delay with low traffic in a static network

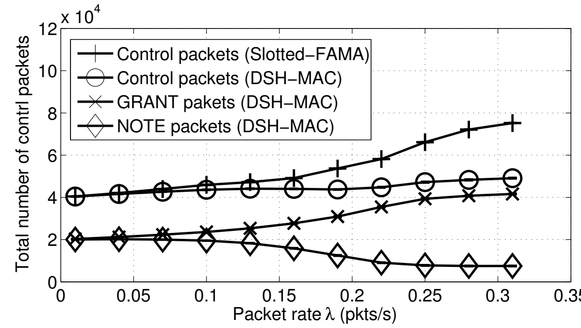

Fig. 8. Average overhead in a static network

theoretical analysis as shown in Fig. 4. The reason that DSH-MAC is able to achieve the highest throughput is twofold. First, with prediction it simplifies the handshaking process, and saves one-way propagation delay for each data transmission, which dominates in the time overhead. Second, with fewer control packets, the channel becomes less congested, and the handshake success rate is higher compared to Slotted-FAMA. Even though with no handshaking overhead, Grant-to-send has a lot of collisions in the network with long propagation delay, leading to retransmissions. The time saved by its grant-to-send mechanism cannot fully compensate the time used for retransmissions due to collisions.

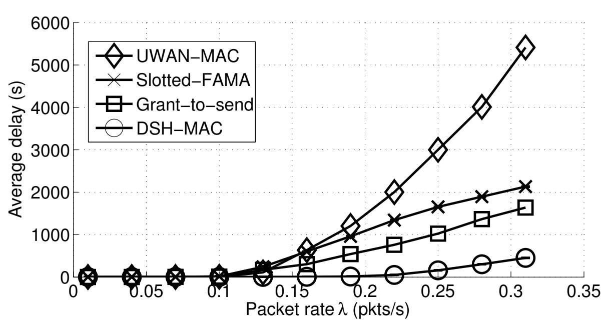

Fig. 7. Average delay in a static network

Fig. 8 compares the control overhead between DSH-MAC and Slotted-FAMA. As Grant-to-send has no dedicated control packets and the overhead of UWAN-MAC is very low, we exclude them from the comparison. In our experiment, every node sends 1000 packets, and $20,000$ packets are sent in total. We show the total number of control packets sent for both DSH-MACand Slotted-FAMA in Fig. 8, and we further show the two different packets - NOTE packets sent without DATA packets and GRANT packets separately for DSH-MAC.

The upper two curves show the total numbers of control packets transmitted by the two protocols. In both curves, the overhead increases as the traffic rate, because more contention failures occur and more control packets are needed. DSH-MAC sends fewer control packets under all the traffic rates and therefore more energy-efficient. The control overhead also affects the throughput. In DSH-MAC, nodes only wait for DATA after a GRANT packet is sent, but are able to schedule other transmissions right after a NOTE packet is sent; whereas in Slotted-FAMA, nodes need to wait after all control packets are sent. By comparing the number of GRANT packets of DSH-MAC to the total number of control packets of Slotted-FAMA, we conclude that DSH-MAC spends less time in waiting.

Next, we evaluate the packet and node fairness of DSH-MAC with Fig. 9 and Fig. 10, respectively. DSH-MAC is better than Slotted-FAMA and Grant-to-send in both fairnesses. The node fairness index of UWAN-MAC is close to one because it strictly gives equal opportunities to each node every cycle. However, a packet can be generated at any time during a cycle and therefore the variance of their queuing delays in UWAN-MAC is large, resulting in lower packet fairness index. The reason of the high fairness in DSH-MAC is that each node grants transmission opportunities to their neighbors based on the total queuing time rather than randomly, and therefore a node with more packets and longer queuing time has more chance to send out its packets. In Slotted-FAMA and Grant-to-send, which node gets chance to send largely depends on random contentions. In Slotted-FAMA, Grant-to-send, and DSH-MAC, as the traffic rate becomes higher, more collisions happen in the network. As a result, which node has opportunity to transmit not only depends on its buffer status, but also depends on certain random factors as well. Therefore, the fairness indices tend to decrease.

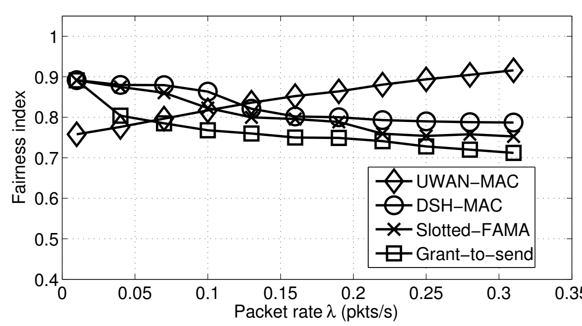

Fig. 9. Packet fairness in a static network

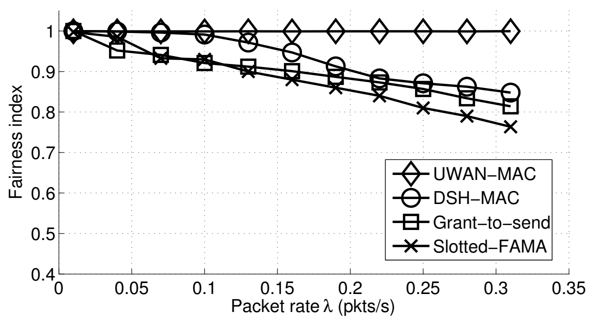

Fig. 10. Node fairness in a static network

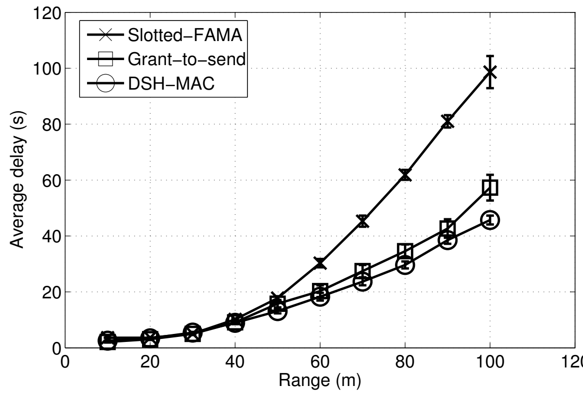

Fig. 11. Average delay change with mobility in a mobile network

C. *Mobile Network*

For a mobile network, robustness is desired in MAC protocols. We demonstrate the adaptiveness of DSH-MAC to the dynamic underwater environment. The movement of nodes may break some links, resulting in handshaking and data transmission failures. We evaluate how such disruptions affect the performance of the protocol in terms of data throughput. Based on the static network above, we set the packet generation rate λ to 0.1 and allow every node to move at 2 m/s on average, causing constant disruptions of the link. Every node vibrates around the original position, and we adjust the maximum mobile range to control the time for a link disruption. As UWAN-MAC takes long time to react to topology changes, it is not included in the comparison.

The curves in Fig. 11 show how the average one-hop delay for the three protocols changes with the mobility range. With the mobility range becomes larger, disconnections between two nodes become more likely while they are performing handshaking and data transmissions, resulting in backoffs and delay. The average packet delay for DSH-MAC grows more slowly compared to the other two protocols as the mobility range increases, indicating that DSH-MAC is more tolerant of link disruptions. Although Grant-to-send has no handshaking process, and data transmission process is shorter, i.e. shorter delay, it has more data packet collisions because it has no handshaking and data packets are much longer than handshaking control packets.

Fig. 12 shows the number of control packets sent (solid curves) and the number of handshaking failures encountered (dashed curves) by DSH-MAC and Slotted-FAMA, with various mobility range. The control overhead for both DSH-MAC and Slotted-FAMA increase as the mobility range rises, due to the handshaking failures. Because DSH-MAC has a shorter handshaking process, it is less likely to have a handshaking failure due to a link disruption, and moreover, the penalty of having a handshaking failure for DSH-MAC is much less in terms of time. As a result, DSH-MAC is able to achieve higher throughput in a mobile network and thus it is more suitable for a dynamic environment.

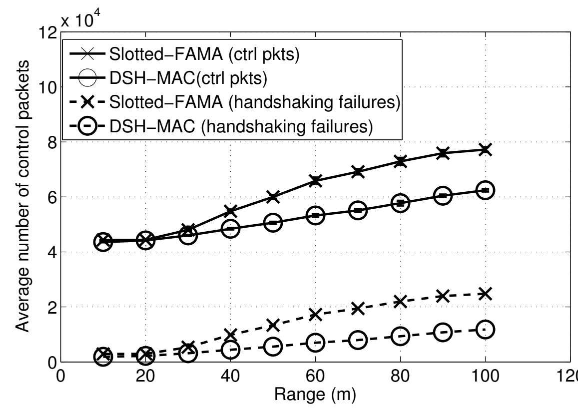

Fig. 12. Total overhead in a mobile network

VI. Conclusion and future work

In this paper, we propose a novel medium access control protocol with decoupled and suppressed handshaking designed specifically for long-delay underwater acoustic networks. Unlike the traditional handshaking with RTS/CTS exchange, the control packets transmissions in the handshaking process of our proposed protocol are decoupled. The benefit is that the sender is able to update the receiver about its demand asynchronously and therefore, the waiting time after the NOTE packet is eliminated. Moreover, with receiver-initiated data communications supported by neighbor queue prediction, the sender does not need to send requests to the receiver every time it desires to transmit, making the handshaking process of our protocol much more efficient.

Both theoretical analysis and extensive experiments have been carried out and demonstrated that our proposed protocol is able to yield higher throughputs with lower overhead at the same traffic rate compared to traditional contention-based protocols.

Our future work includes future shorten the handshaking process, possibly by combining some of GRANT packets with data packets.

Acknowledgement

This work was supported by the Maritime Sensing program of Office Naval Research under grants N00014-10-1-0264 and N00014-12-1-0070.

References

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, “Underwater acoustic sensor networks: Research challenges,” *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, May 2005.

[2] Y. Ma, Z. Guo, Y. Feng, M. Jiang, and G. Feng, “C-MAC: A TDMA-Based MAC Protocol for Underwater Acoustic Sensor Networks,” in *Proc Intl. Conf. on Networks Security, Wireless Communi. and Trusted Computing*, Apr 2009, pp. 728–731.

[3] M. K. Park and V. Rodoplu, “UWAN-MAC: An energy-efficient MAC protocol for underwater acoustic wireless sensor networks,” *IEEE J. Oceanic Engineering*, vol. 32, no. 3, pp. 710–720, 2007.

[4] P. Xie and J.-H. Cui, “R-MAC: An energy-efficient MAC protocol for underwater sensor networks,” *Int. Conf. on Wireless Algorithms,Systems & Applications*, pp. 187–198, Aug. 2007.

[5] J. Yackoski and C.-C. Shen, “UW-FLASHR: Achieving high channel utilization in a time-based acoustic MAC protocol,” in *Proc. WUWNet*, Sept. 2008, pp. 59–66.

[6] M. Molins and M. Stojanovic, “Slotted FAMA: A MAC protocol for underwater acoustic networks,” in *Proc. IEEE Oceans Conf.*, May 2006, pp. 1–7.

[7] X. Guo, M. Frater, and M. Ryan, “A propagation-delay-tolerant collision avoidance protocol for underwater acoustic sensor networks,” in *OCEANS*, May 2007, pp. 1–6.

[8] A. Syed, W. Ye, and J. Heidemann, “T-Lohi: A new class of MAC protocols for underwater acoustic sensor networks,” in *Int. Conf. on Computer Communi.*, Apr. 2008, pp. 231–235.

[9] Z. Peng, Y. Zhu, Z. Zhou, Z. Guo, and J.-H. Cui, “COPE-MAC: A Contention-based medium access control protocol with Parallel Reservation for underwater acoustic networks,” in *OCEANS*, May 2010, pp. 1 –10.

[10] Y.-D. Chen, S.-S. Liu, C.-M. Chang, and K.-P. Shih, “CS-MAC: A Channel Stealing MAC protocol for improving bandwidth utilization in underwater wireless acoustic networks,” in *OCEANS*, Sept. 2011, pp. 1 –5.

[11] C. Petrioli, R. Petroccia, and M. Stojanovic, “A comparative performance evaluation of MAC protocols for underwater sensor networks,” in *Proc. OCEANS*, 2008, pp. 1–10.

[12] J. I. Choi, M. Jain, M. A. Kazandjieva, and P. Levis, “Granting silence to avoid wireless collisions,” in *Proc. IEEE ICNP*, 2010, pp. 82–91.

[13] Andras Varga and OpenSim Ltd., *OMNeT++ User Manual*, 2010. [Online]. Available: http://www.omnetpp.org

[14] *INET Framework Manual*, 2012. [Online]. Available: http://inet.omnetpp.org

[15] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, “A quantitative measure of fairness and discrimination for resource allocation in shared computer systems,” DEC-TR-301, Digital Equipment Corporation, Tech. Rep., Sep 1984.