This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

*Digital Object Identifier 10.1109/ACCESS.2017.DOI*

# Delay and Queue Aware Adaptive Scheduling-Based MAC Protocol for Underwater Acoustic Sensor Networks

**XIAOXIAO ZHUO[1], FENGZHONG QU[1], (Senior Member, IEEE),HONG YANG[1], YAN WEI[1], YEZHOU WU[1], AND JIANGHUI LI[3]**
[1]Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Zhejiang University, Zhoushan 316021, China
[2]Institute of Sound and Vibration Research, University of Southampton, UK; J.Li@soton.ac.uk

Corresponding author: Fengzhong Qu (e-mail:jimqufz@gmail.com).

This work was supported in part by the National Natural Science Foundation of China for Excellent Young Scholars under Grant 61722113; the Joint Program of National Natural Science Foundation of China and Zhejiang Province under Grant U1809211; the Foundation of National Defense Science and Technology Key Laboratory under Grant 614210903060717.

**ABSTRACT** This paper mainly investigates the application of underwater acoustic sensor networks (UASNs), which are characterized by large-scale, sparse distribution, and varying traffic loads. Since the underwater acoustic channel is known for limited bandwidth, time variation, and long propagation delays, the strategy to access the common communication medium is required to improve the performance of UASNs. This paper proposes a delay and queue aware adaptive scheduling-based medium access control (DQA-MAC) protocol for UASNs. It combines adaptive scheduling transmission, reduction of handshaking packets, and concurrent transmission with the propose of improving the performance of network throughput, shortening end-to-end delay, reducing average energy consumption, and enhancing the fairness of transmission. Data transmission time is scheduled based on the information of propagation delays and the number of data packets waiting in each node queue. Furthermore, the strategy of concurrent transmission is implemented to leverage the long propagation delays. At last, reducing the number of handshaking packets is achieved with the approach of exchanging information by specially designed packets frames. The simulation results show that the proposed protocol outperforms the related traditional protocols in networks with varying traffic loads.

**INDEX TERMS** Underwater acoustic sensor networks, propagation delay, medium access control, collision avoidance, concurrent transmission

## I. INTRODUCTION

In the past few years, underwater acoustic sensor networks (UASNs) have drawn wide attention due to their broad applications in marine exploration, such as oceanographic data collection, pollution monitoring, tsunami warning [1], [2], etc. However, the volatile ocean environment makes underwater acoustic signals propagation suffer from many challenges in UASNs, such as limited bandwidth, low data rate, and low propagation speed [3]. These features make it an unprecedented challenge to design UASNs, such as low network throughput, long end-to-end delay, and large energy consumption. In addition, it is well-known that medium access control (MAC) protocols, which allow multiple users to share the common medium fairly and efficiently, are important for the performance of UASNs. Much effort has been paid to design an appropriate protocol for a specific UASN.

This paper mainly focuses on the design of the MAC protocol for the sensor networks with multiple centralized networks. In each centralized networks, the sensor nodes (SNs) can reach the central node (CN) in only one hop. And the CN is functioning as a data sink to interface with the base station. These networks have features of long propagation delays, varying traffic load, and sparsely distributed nodes. 1) Long propagation delays are caused by the slow sound speed in ocean environment only at 1500 meters per second. On the one hand, this feature results in long round-trip time (RTT) in UASNs. Therefore, the scheme of exchanging the request-to-send (RTS)/clear-to-send (CTS) packets before data packets transmission which is designed in traditional

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

protocols becomes expensive. On the other hand, leveraging the propagation delays as an opportunity instead of being constrained by it seems to be a promising strategy, which allows the concurrent transmission to occur with a collision-free reception. 2) In UASNs, the traffic load in each node is dynamic and large, which incurs data congestion. Therefore, it is necessary to pay close attention to the numbers of data packets waiting in node queues. 3) To monitor the immense ocean area with less cost, nodes in UASNs are supposed to be sparse and separated.

To cater to the above specified features, we propose a MAC protocol named delay and queue aware adaptive scheduling-based MAC protocol (DQA-MAC). To avoid packets collisions, in each round this protocol first updates the information of propagation delays and the number of data packets in each node queue and then schedules the packets transmission time. Furthermore, to improve channel utilization, it reduces the handshaking packets and exchanges information by special designed packets frames to reduce the number of control packets. At last, it leverages the long propagation delays to implement concurrent transmission to improve the network throughput. Therefore, the contributions of our research are led to, which are:

1) In the proposed MAC protocol, the strategies of adaptive scheduling transmission, reduction of handshaking packets and concurrent transmission are combined to improve the performance of the network.
2) In the proposed protocol, by applying the information of propagation delays and the number of data packets waiting in each node queue to scheduling data transmission time, the fairness of the network could be enhanced.
3) The network throughput of the proposed MAC protocol is analyzed, and the performances of network throughput, end-to-end delay, and energy consumption are simulated and compared with the related MAC protocols as well.

The remainder of this article is structured as follows. Section 2 discusses the related work about this research. The details about the proposed DQA-MAC protocol are presented in Section 3, followed by the approximate performance analysis in Section 4. In Section 5, the numerical simulation results are compared and the discussions are given. At last, concluding remarks are made in Section 6.

## II. RELATED WORK

Several existing MAC protocols have been developed for UASNs and can mainly be divided into two categories: contention-based and schedule-based protocols. In contention-based protocols, sensor nodes compete for the opportunity of channel accessing with methods of random access or handshaking. In terms of random access methods, some protocols such as ALOHA [4], slotted-ALOHA [5] and CSMA [6] have been modified from the terrestrial MAC protocols and become suitable for the UWANs. In these protocols, nodes transmit data packets or sensing carriers at anytime according to their own will. And with no handshaking, they make the end-to-end delay of the network small. However, in random access protocols, with the increase of the traffic load, there exist the hidden and exposed terminal problems. This leads to the increase of packets collisions and the decrease of the network throughput accordingly. Thus, the handshaking protocols are investigated to handle these problems. The most well-known handshaking protocols are multiple access collision avoidance (MACA) [7] and Slotted-FAMA [8]. In MACA, nodes have to send RTS/CTS packets in advance to reserve the channel before transmitting data packets and then an acknowledge (ACK) packet is replied to inform the node if the data packet have been successfully received. In this way, one data packet has to go through a four-way handshaking, which inevitably decreases the network throughput due to the long propagation delay and long RTT. Moreover, slotted-FAMA is subsequently designed to divide time into slots and packets are allowed to send only at the beginning of each time slot. Furthermore, some protocols take actions to achieve a higher throughput. The adaptive propagation-delay-tolerant collision-avoidance protocol (APCAP) [9] leverages long propagation delays and sets a defer time for the CTS and DATA packets transmission, which allows nodes to ensure that packets can reach the destination nodes without collision. The defer time also allows nodes to transmit data packets or handshaking packets in the next round. To save the transmission time of control packets, the bidirectional-concurrent MAC (BIC-MAC) [10] and Twin-TDMA [11] protocols allow senders and receivers of the RTS packets to send data packets to each other at the same time slot. The idea of the MACA-MN [12] protocol is that one RTS packet of the sender is designed to require for multiple data packets instead of only one to be sent to its neighbors to improve the handshaking efficiency. The prerequisite of this method is to design a collision-avoidance mechanism of replied multiple CTS packets. In addition, the MACA-U [13] and FI-MACA [14] are also modified based on MACA protocols to improve the chance of successful handshaking and further enhance the network throughput.

The scheduling-based protocols are popular in UASNs. Most of them is designed to utilize the information of the network to schedule the packets transmission time to avoid collisions or implement concurrent transmission to improve the network throughput. Among them, some are sender-oriented protocols. Most of them collect some additional information about the network to avoid collisions. The ALOHA-CA [15] first overhears every frame to obtain the information of sender and receiver and then schedule the packets transmission with the information of propagation delays to avoid collisions. The spatial-temporal (ST-MAC) [16] collects the information of traffic loads and routing to construct a collision-avoidance and fairness transmission. The STUMP [17] utilizes the information of estimated propagation delays and the schedules of neighborsâĂŹ transmission to prevent collision and save the channel idle time as much as possible. To improve the network throughput, concurrent

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

transmission is developed. The authors in [18] establish the collision avoidance constrain model with a mixed integer linear programming based on the information of routing and propagation delays. This strategy takes advantage of the long propagation delay and implement concurrent transmission. In the propagation delay-aware opportunistic (DOTS) [19] MAC protocol, nodes are allowed to overhear the information of neighboring transmissions and then build a delay map to support concurrent transmission. And some receiver-oriented protocols are investigated as well. In the handshake-based ordered scheduling MAC (HOSM) [20] protocol, the senders are ordered and scheduled by the receivers. It allows multiple senders to send data packets in one handshaking round to improve the handshaking efficiency. For the receiver-initiated packet train MAC protocol (RIPT) [21], to increase bandwidth utilization and avoid packets collisions, the receivers reserve the channel for the sender on the promise that every node should know the inter-node propagation delay between itself and each of its adjacent neighbors. In addition, the DCO-MAC [22] implements a receiver-based MAC protocol in heavy traffic load and a contention-based MAC protocol in light traffic load with the expectation of improving the network throughput and decreasing the network delay. The asymmetric propagation delay-aware TDMA (APD-TDMA) [23] MAC protocol addresses the asymmetric propagation delay in mobile UASN, and in this protocol, the receiver requires for data packets periodically and collision at the receiver is avoided by deferring data packet transmission after reception of a beacon packet from the SN. Furthermore, the TDA-MAC protocol [24] provides TDMA and concurrent transmission to sensor node by sending a request (REQ) packets without the need for centralized clock synchronization. This protocol is capable of matching the performance of an ideal staggered TDMA protocol.

In both of the above two main categories, there exist some protocols exploring the fairness of the network. Fairness means that all senders transmit almost the same amount of data packets to the receiver in the long term. The spatially fair MAC (SF-MAC) [25] adopts a receiver-based packet transmission without considering the distance information. The receiver gets a number of RTS packets from the senders and then sends the CTS packet back to the sender of which the RTS packet arrives earliest. Moreover, a heuristic algorithm is elaborated to determine the optimal frame length of a STDMA [26] UASN with a fairness requirement.

In the protocols mentioned above, the RTS/CTS mechanism is widely used. Although this mechanism solves the problems of packets collision and hidden/exposed terminal in UASNs, it introduces low channel utilization and leads to low throughput and long end-to-end delay due to long propagation delays and long RTT. Therefore, the DQA-MAC we proposed decides to abandon the RTS/CTS mechanism and takes action of ordering and scheduling the packets transmission time to avoid collisions. In addition, it realizes concurrent transmission of multiple senders to improve the network throughput. With regard to the fairness of the network, the protocol orders the nodes to transmit data packets based on the information of propagation delays and the data numbers waiting in queues, which has not been adopted in the existing protocols. In this way, the protocol improves the network throughput, decreases the end-to-end delays, reduces the energy consumption, and enhances the fairness of the network.

## III. DQA-MAC PROTOCOL DESIGN

### *A. NETWORK MODEL*

In this paper, we mainly consider a typical, non-mobile underwater acoustic sensor network with traffic-varying load, which is designed to sense ocean information and forward it to a data center via a sink node. As shown in Fig. 1, the network consists of multiple star networks, where SNs collect sensing data and transmit to a CN. Some assumptions are made in this network model. First, each sensor node works in half-duplex mode, which means that transmission and reception could not happen simultaneously. Second, the SNs are all located within the one-hop communication range of the CN. Third, SNs are synchronized in time with the center nodes. This could be achieved with the method in [27], which relies on the opportunistic information of timestamps that can be obtained in the proposed DQA-MAC protocol. In this way, the clock drift can be reduced to 0.15ms/h for a duration of 4h and the accuracy of this time synchronization is acceptable for our target network. Fourth, due to the sparse distribution of the SNs in UASNs, it is assumed that the adjacent star networks would not interfere with each other. At last, all of the SNs are static or slightly drift with the ocean current around their fixed position.

### *B. HOW THE DQA-MAC PROTOCOL WORKS*

The proposed DQA-MAC protocol is a delay-queue aware and receiver-oriented MAC protocol. To achieve high network throughput, the protocol reduces handshaking packets and implements concurrent transmission within one round. To avoid concurrent packets collisions, receivers schedule the data packets transmission time. A sender transmits a packet according to the transmission time arranged by its receiver. Considering the fairness of the network, the scheduling rule of the data packets transmission is based on the algorithm named the Exponential Rule [29]. It schedules the order of transmission based on the information of queue length and delay. If we schedule the order of the SNs only based on the information of propagation delays, it will fall into the situation that only nearest node can access the channel first and the nodes far away and in heavy traffic may lose their data packet due to the long waiting time. Thus, to improve the fairness of the network, we employ the exponential rule to schedule the order of the SNs. In addition, The information of transmission time, propagation delays, and queue length could all be obtained by the exchange of packets. Their frame formats are specially designed and described in Fig. 3.

There are four phases in the protocol and an example of its description is presented in Fig. 2. The frame formats of

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

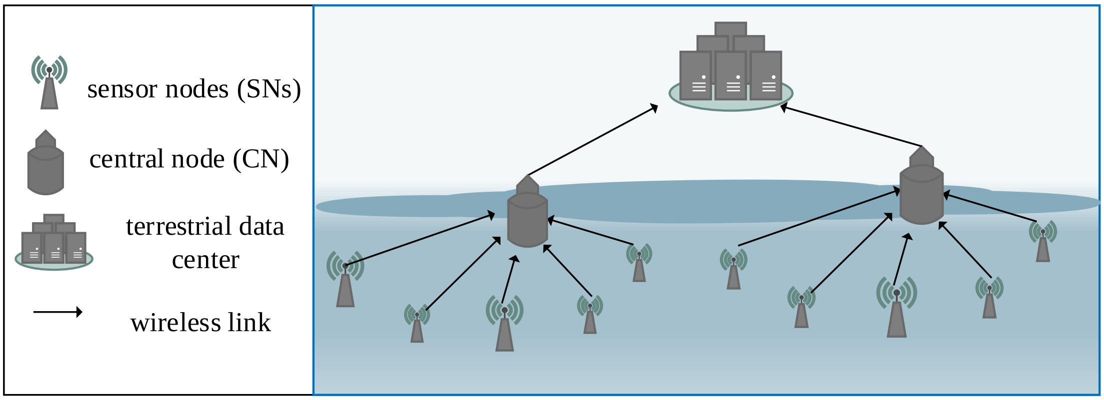


FIGURE 1: Network model

packets are specially designed and described in Fig. 3.

1) The initial phase is to initiate the information of propagation delays and queue length. All SNs send initial request sending (IRES) packets containing transmission timestamps and queue lengths to the CN as shown in Fig. 3.
2) After that, an initial scheduling phase starts aiming to initiate the first data packets scheduling. In this process, after collecting all the IRES packets from the SNs, the CN creates a scheduling table, which stores the MAC addresses of senders and their corresponding propagation delays and queue length. And then the CN sets up a waiting transmission time table to calculate the waiting time of data packets transmission for each SN. How to build the scheduling table and waiting transmission time table will be detailed afterward. After this operation, the CN sends an initial scheduling (ISP) packet to inform SNs the "waiting time" to transmit the data packets. Note that the "waiting time" is included in the ISP packet as described in Fig. 3.
3) Then the process proceeds to the data transmission phase. After receiving the ISP packet, the SN waits according to the "waiting time" specified by the ISP packet and then sequentially sends the data packet. It is worthy of mentioning that the transmission timestamps and queue lengths are contained in the data packets as shown in Fig. 3.
4) Finally, in the rescheduling phase, when the CN receives all the data packets, it will update both the scheduling table and the waiting transmission time table and send an acknowledgement (ACK) packet to inform the SNs if it has received the data packet successfully (indicated by the ACK symbol in Fig. 3) and when to send the data packet in the next round (represented by the waiting time). If the data packet is successfully received, the SN will send the next data packet in the next round. Otherwise, the sender will retransmit the data packet. The rescheduling phase ends up. The initial phase and initial scheduling phase are implemented only in the first round. In the rest of the process, nodes will repeat the data transmission phase and the rescheduling phase.

More details about the proposed DQA-MAC protocol will be elaborated in the remainder of this section. First, a scheme is put forward to ensure the successful reception of IRES packets in the initial transmission phase. Second, how to establish the scheduling table will be explained. Third, the process of setting up the waiting transmission table will be presented, followed by the details of updating the waiting transmission time table at last.

### C. IRES PACKETS RETRANSMISSION

In the initial phase, all SNs send IRES packets to the CN when they want to send the data packet. Since this is a random access process, packets collisions may occur and the IRES packets could not be successfully received. Once this occasion appears, the CN could not collect the information of all SNs and the process of the protocol would not proceed. So, to ensure that the IRES packets are successfully received, we retransmit the IRES packets after a random backoff time until receiving the ISP packet. The backoff time is set to be large than the data transmission time to reduce packets collisions. The pseudo code of this method is shown in Algorithm 1.

### D. ESTABLISHING THE SCHEDULING TABLE

In the process of the protocol, a SN i will send the IRES or DATA packets containing transmission timestamp $t_{si}(n)$ and queue length $q_i(n)$. When the CN receives the IRES or DATA packets, it will record the receiving time $t_{ri}(n)$. The propagation delay can be denoted by

$$\tau_i(n) = t_{ri}(n) - t_{si}(n) \tag{1}$$

After receiving all packets from the SNs in the same star network, every CN sets up a scheduling table for the SNs, as shown in Table 1, which contains MAC addresses of the

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

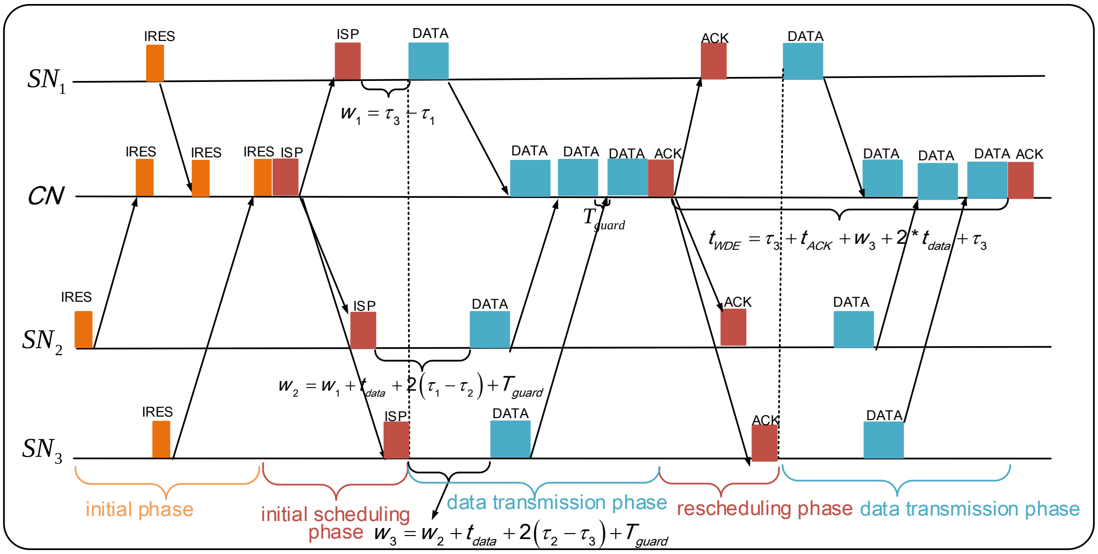


FIGURE 2: An example of the packet transmission process of DQA-MAC

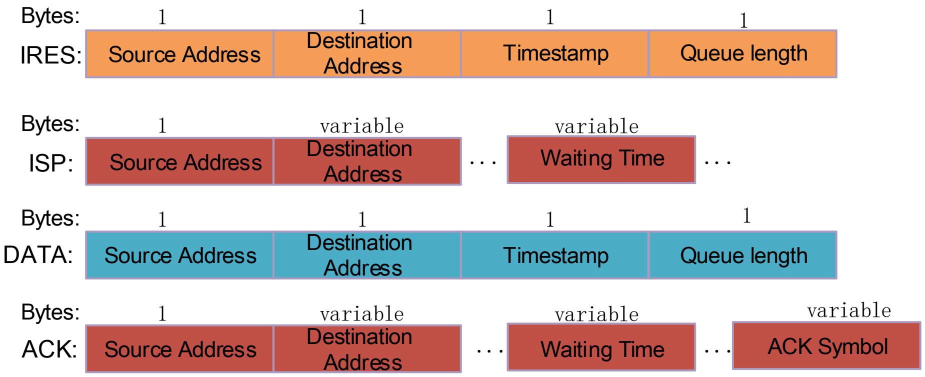


FIGURE 3: Frame formats of packets

SNs and their corresponding queue lengths and propagation delays.

TABLE 1: Scheduling table

| SN | MAC Address | Sending Time | Receiving Time | Propagation Delay | Queue Length |
|---|---|---|---|---|---|
| 1 | xx-xx-xx-x1 | $t_{s1}$ | $t_{r1}$ | $\tau_1$ | $Q_1$ |
| 2 | xx-xx-xx-x2 | $t_{s2}$ | $t_{r2}$ | $\tau_2$ | $Q_2$ |
| 3 | xx-xx-xx-x3 | $t_{s3}$ | $t_{r3}$ | $\tau_3$ | $Q_3$ |
| ... | ... | ... | ... | ... | ... |

### E. SETTING UP THE WAITING TRANSMISSION TIME TABLE

Setting up the waiting transmission time table can be divided into two steps. The first step is to determine the order of DATA packets transmission, and the second is to calculate the waiting time for every SN to transmit DATA packets.

The order of transmitting data packets of the SNs is associated with both the propagation delays $\tau_i(n)$ and the queue lengths $q_i(n)$, subject to the exponential (EXP) rule scheduler. Based on the exponential function, this scheduler orders the SN of which delay and queue length would be weighted. The exponential rule is given by

$$index_i(n) = \frac{\gamma_i(Q_i(n) - Q_i(n-1))}{t_{ri}(n) - t_{ri}(n-1)} exp \frac{\alpha_i \tau_i(n) - \frac{1}{N} \sum_{i=1}^{i=N} \alpha_i \tau_i(n)}{1 + \sqrt{\frac{1}{N} \sum_{i=1}^{i=N} \alpha_i \tau_i(n)}}, \quad (2)$$

where $n$ is the $nth$ round, N is the number of SNs in a star network. $index_i(n)$ is the exponential result of the ith SN, and $Q_i(n)$ is the predicted queue length waiting in the ith SN when the CN sets up a waiting transmission time table. $Q_i(n)$ is denoted by

$$Q_i(n) = Q_i(n-1) - 1 + \lambda_i(n) * (t_{ri}(n) - t_{ri}(n-1)), \quad (3)$$

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

**Algorithm 1** IRES packets retransmission

```
if the first time to send IRES then
    send IRES packet and set backoff time
    set status SEND_IRES and set txtime
    if txtime expires then
        set status IRES_WAIT_ISP
    end if
end if
while backoff time expires do
    if the status is IRES_WAIT_ISP then
        send IRES packet and set backoff time
        set status SEND_IRES and set txtime
        if txtime expires then
            set status IRES_WAIT_ISP
        end if
    else
        if receive the ISP packet then
            set status SEND_DATA and stop backoff timer
            send DATA packet after the waiting transmission time
        end if
    end if
end while
```

where $\lambda_i(n)$ is the packet arrival rate at node $i$. The SNs are ordered based on $index_i(n)$ sorted in an ascending order and an assumption is made that the $jth$ order to transmit packet is the SN $O_j(n)(j \in 1, , N)$. For example, if the $index_2(n)$ is in the first place, $j$ will be 1 and $O_1(n)$ is 2.

The waiting time for every SNs is the time between receiving the ISP or ACK packets and sending a DATA packets. More specifically, the SN in the first place allowed to transmit the DATA packet only when all of the SNs, including the farthest one, receive the ISP or DATA packets. To avoid packets collisions, DATA packets transmissions are scheduled to arrive at the CN one by one with a short guard time $T_{guard}$. The example of the protocol shown in Fig. 2 assists in understanding this mechanism. Assume that a data packet length is $t_{data}$ and the maximal propagation delay is $\tau_{max}$. The waiting time $w_1$ of the first ordered node is expressed as:

$$w_{O_1}(n) = \tau_{max}(n) - \tau_{O_1}(n), \tag{4}$$

which means that SN $O_1$ will transmit its data packet after $w_1(n)$ expiring. And the waiting time for the jth order node SN $O_j$ is

$$w_{O_j}(n) = w_{O_{j-1}}(n)+t_{data}+2*max(\tau_{O_{j-1}}(n)-\tau_{O_j}(n),0)+T_{guard}. \tag{5}$$

In this way, the protocol not only implements concurrent transmission but avoids collisions. Hence, the waiting transmission time table is set up as shown in Table 2. The pseudo code of this method is shown in Algorithm 2.

TABLE 2: Waiting tansmission time table

| SN | MAC Address | Waiting Delay |
|---|---|---|
| 1 | xx-xx-xx-x1 | $w_1$ |
| 2 | xx-xx-xx-x2 | $w_2$ |
| 3 | xx-xx-xx-x3 | $w_3$ |
| ... | ... | ... |

**Algorithm 2** Setting up the scheduling table and the waiting transmission time table

```
if the CN receives all the IRES packet then
    record the receiving time $t_{ri}(n)$
    calculate the propagation delay $\tau_i(n) = t_{ri}(n) - t_{si}(n)$
    store the MAC addresses of the SNs and their corresponding queue lengths and propagation delays into the scheduling table
end if
for all $i \in N$ do
    calculate $index_i(n)$
end for
for all $i, j \in N$ do
    sort $index_i(n)$ via the sorting algorithm and get the corresponding $O_j$
end for
$waiting\ time_{O_1}(n) = \tau_{max}(n) - \tau_{O_1}(n)$
for all $i, j \in N$ do
    calculate the waiting time $w_{O_j}(n) = w_{O_{j-1}}(n) + t_{data} + 2 * max(\tau_{O_{j-1}}(n) - \tau_{O_j}(n), 0) + T_{guard}$
end for
```

### *F. UPDATING THE SCHEDULING TIME TABLE*

In some cases, the DATA packets may not be received correctly resulting in the information of transmission timestamp $t_{si}(n)$ and queue length $q_i(n)$ missed. To cope with this, three tips are adopted in this protocol. First, a waiting end time (WDE) $t_{WDE}$ is set to end the data transmission phase no matter whether all of the DATA packets have been received. The WDE time is sufficient enough for a round of data transmission phase and the rescheduling phase to guarantee the reception of all the DATA packets. Moreover, the WDE time prevents CN from the endless waiting in the data transmission phase. $t_{WDE}$ is calculated as:

$$t_{WDE}(n) = \tau_{O_N}(n)+t_{ACK}+w_{O_N}(n)+2*t_{data}+\tau_{O_N}(n), \tag{6}$$

where $t_{ACK}$ is the ACK packets transmission time.

The second tip is to supplement the missing information of propagation delay and the queue length. Since the propagation delay may be not changed much in a round, it is acceptable to take the propagation delay of $(n-1)th$ round as $nth$ round, i.e.

$$\tau(n) = \tau(n-1). \tag{7}$$

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

And the queue length in $nth$ round is predicted based on the traffic load and the queue length in $(n-1)th$ round, and denoted by

$$Q_i(n) = Q_i(n-1) + \lambda_i(n) * (t_{ri}(n-1) - t_{ri}(n-2)). \quad (8)$$

And then, the CN will update the waiting transmission time table based on $\tau(n)$ and $Q_i(n)$. At last, the frame format of ACK packets contains an acknowledgment symbol to inform the SNs if its DATA packet has been received: 1 indicates yes and 0 no. The pseudo code of this method is shown in Algorithm 3.

**Algorithm 3** Updating the scheduling time table

```
if the CN sends the ACK or ISP packet then
    turn on the waiting end time $t_{WDE}(n) = \tau_{O_N}(n) + t_{ACK} + w_{O_N}(n) + 2 * t_{data} + \tau_{O_N}(n)$
end if
while the CN receives the DATA packet from a SN do
    record the receiving time $t_{ri}(n)$
    calculate the propagation delay $\tau_i(n) = t_{ri}(n) - t_{si}(n)$
    store the MAC addresses of the SN and its corresponding queue length and propagation delay into the scheduling table
    if the waiting end time expires then
        if the CN have not receives the DATA packet of SN $i \in N$ then
            $\tau(n) = \tau(n-1), Q_i(n) = Q_i(n-1) + \lambda_i(n) * (t_{ri}(n-1) - t_{ri}(n-2))$
            store the MAC addresses of the SN $i$ and its corresponding queue length and propagation delay into the scheduling table
        end if
    end if
end while
```

## IV. APPROXIMATE PERFORMANCE ANALYSIS

In this section, the network throughput performance of our protocol is analyzed. It is assumed that there are N nodes in the network and each node follows a Poisson distribution with average $\lambda$ packets per second. According to the flow model in [30], the channel utilization can be defined as the ratio of average data transmission time and total transmission time, which is given by

$$S = \frac{\bar{U}}{\bar{B} + \bar{I}}, \quad (9)$$

where $\bar{U}$ denotes the expected time for transmitting data packets, $\bar{B}$ denotes the expected time when the channel is busy, and $\bar{I}$ denotes the expected time when the channel is idle. The busy period mainly contains the data transmission phase when there are data packets to transmit and rescheduling phase. The idle period is the duration that no data packet is transmitted in all nodes. Their formulas will be given later.

Since the initial phase and initial scheduling phase are proceeded only once and are rather small compared to the whole process, it is regarded reasonable to ignore them in our analysis. Since the order and the propagation delay of the first node vary based on the information of queue length and delay. So they are uncertain in each round of the protocol in the real scenario. So to simplify the analysis, we define the propagation delay of the first node as $\tau_{max}$ whatever is the first node to transmit data. Therefore, the successful data transmission period is composed of the successful transmission of the DATA packets and ACK packets. The successful DATA packets transmission period can be calculated as follows:

$$\begin{aligned} &T_{DATA} \\ &= \tau_{max} + t_{DATA} + \bar{K} \sum_{i=0}^{\infty} i(t_{DATA} + t_{guard}) P_e^{i-1}(1-P_e) \\ &= \tau_{max} + \bar{K} * \frac{t_{DATA} + t_{guard}}{1-P_e}, \quad (10) \end{aligned}$$

and the ACK packets transmission period can be denoted by

$$T_{ACK} = \tau_{max} + t_{ACK}. \quad (11)$$

Thus, the period of succeful data transmission period is given by:

$$\begin{aligned} T_{SUC} &= T_{DATA} + T_{ACK} \\ &= 2\tau_{max} + t_{ACK} + t_{DATA} + \frac{\bar{K} * (t_{DATA} + t_{guard})}{1-P_e}, \quad (12) \end{aligned}$$

where $P_e$ is the packet error rate and $t_{guard}$ is the guard time in the data transmission phase. $t_{DATA}$ and $t_{ACK}$ are the transmission delay of DATA packet and of ACK packet, respectively. $\bar{K}$ is the number of DATA packets transmitted in the network during a round and can be calculated as follows:

$$\bar{K} = \sum_{K=1}^{N-1} k * P^k, \quad (13)$$

where $P$ denotes the probability of generating data packets during a round, which is

$$P = 1 - e^{-\lambda T_{round}}. \quad (14)$$

where $T_{round}$ is the period of a round and denoted by

$$T_{round} = t_{DATA} + \tau_{max} + t_{ACK} + \tau_{max} + N * (t_{DATA} + t_{guard}). \quad (15)$$

Following this, the probability of generating data packets in a round is

$$P^k = C_N^k P^k (1-P)^{N-k}. \quad (16)$$

Substituting (16) into (13), we can get

$$\bar{K} = \sum_{k=1}^{N-1} k * P^k = \sum_{K=1}^{N-1} k C_N^k P^k (1-P)^{N-k} = NP. \quad (17)$$

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

The failed transmission period $T_{FAIL}$ means no data packets to transmit and is denoted by

$$T_{FAIL} = (N - \bar{K}) * (t_{DATA} + t_{guard}). \tag{18}$$

Since the channel is always occupied when there are data packets to transmit and only when no data packet is transmitted in all node, the channel becomes idle. Therefore, the expected time when the channel is idle is given by:

$$\bar{I} = T_{FAIL}. \tag{19}$$

Furthermore, the busy period is a successful transmission process in which there is at least one data packet to transmit. It can be calculated as:

$$\bar{B} = T_{SUC}, \tag{20}$$

The expected time for transmitting data packets is given as follows:

$$\bar{U} = \bar{K} * t_{DATA}. \tag{21}$$

Therefore, substituting (19), (20), and (21) into (9), the channel utilization can be calculated as:

$$S = \frac{\bar{K} * t_{DATA}}{T_{SUC} + T_{FAIL}}. \tag{22}$$

From the equation, we can find that the throughput of DAQ-MAC is determined by $\lambda$, N, $\tau_{max}$, $t_{DATA}$, and $t_{ACK}$.

## V. SIMULATION RESULTS AND DISCUSSIONS

### A. SIMULATION SETUP

We conduct simulations to demonostrate the performace of the proposed DQA-MAC protocol. Simulations are carried out based on Aqua-Sim, which is an underwater sensor network simulation extension package based on Network Simulator 2 [31]. The settings of simulations are introduced as follows. In the network, there are N nodes, which are all randomly deployed in a 3000m×3000m×3000m cube, and the CN is at the center. Data packets in each node are generated following a Poisson process with an average rate of $\lambda$. Unless otherwise mentioned, all data packets are 50 bytes. The MAC header of DATA and IRES packets is 4 bytes and the MAC header of ISP and ACK packets varies along with the number of nodes in the network. The guard time is 0.01s. In the physical layer, we use an error-free channel model where sound wave propagates at the speed of 1500 m/s so that the missed packets are mainly caused by collisons. The data transmission rate is set at 1kbps, and the maximum transmission range is 3000m. Power settings are 10 W for transmission, 1 W for reception, and 0.2 W in idle state.

We compare the DQA-MAC protocol with the representative MAC protocols, i.e. HOSM [20], UWALOHA_NO_ACK [4], and Slotted-FAMA [8]. The HOSM is a receiver-oriented scheduling-based MAC protocol with the strategies of concurrent transmission, scheduling packets transmission time and two-way handshaking. It is used to compare with the DQA-MAC to demonstrate the latter's advantages of reduction in handshaking packets. The UWALOHA is the baseline of a random access MAC protocol for comparison to verify that our proposed protocol possesses the advantage of dynamically scheduling packets transmission. And the Slotted-FAMA is handshaking based MAC protocol to show the advantage of concurrent transmission and reduction of handshaking packets considered in the DQA-MAC protocol.

The performances of network throughput, end-to-end delay, average consumed energy, and fairness index are all evaluated with different numbers of nodes under varying traffic loads.

1) Network throughput, indicating the channel utilization in the simulation time, is defined as the number of packets that are successfully received per second, i.e.

$$Throughput = \frac{Number\ of\ data\ packets\ received}{Simulation\ time}. \tag{23}$$

2) End-to-end delay is the average time between a data packet released from the sender and received from the receiver, i.e.

$$End-to-end\ delay = \frac{Total\ delay}{Number\ of\ received\ data\ packets}. \tag{24}$$

3) Average energy consumption is the average consumed energy of successful delivered data packets per second, i.e.

$$Consumed\ energy = \frac{Total\ energy}{Number\ of\ received\ data\ packets}. \tag{25}$$

4) Fairness index affects the network survival time which is critical for UASNs and represents the fairness of the chance to access channel. Here, with the application of Jain's fairness index (FI) [32], the fairness of MAC protocols is defined as follows:

$$FI = \frac{\left(\sum_{i=1}^{N} m_i\right)^2}{N \times \sum_{i=1}^{N} m_i^2}, \tag{26}$$

where N denotes the number of nodes, and $m_i$ is the number of data packets successfully received from sender $i$. The value of FI ranges from 0 to 1 and 1 means the best performance in fairness index, indicating that the number of data packets received from all of the SNs is the same.

Generally speaking, a well-desinged MAC protocol is expected to have high network throughput, short end-to-end delay, low consumed energy, and good fairness index.

### B. SIMULATION RESULTS

#### 1) Throughput

As shown in Fig. 4, the network throughput is investigated with the traffic load varying from 0.1 to 1 pkt/s. It is observed that in general the throughputs of the protocols climb up along with the growing traffic loads and all of them have

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

the upbound. The number of nodes joining in the network in Fig. 4(a) and Fig. 4(b) are 6 and 11, respectively. These two figures display that the network throughput of proposed DQA-MAC protocol is better than the other protocols. More specifically, the DQA-MAC performs better than UWALOHA_NO_ACK, indicating that the scheduling based transmission outperforms random access approaches. Moreover, the performance of DQA-MAC and UWALOHA_NO_ACK are better than the Slotted-FAMA and HOSM, suggesting that the strategy of reduction of handshaking packets is useful in this network model. Also, the performance of the HOSM is better than the Slotted-FAMA demonstrating that the concurrent transmission is effective.

In addition, the theoretical throughput of the proposed DQA-MAC is evaluated and can be calculated from (22) as follows,

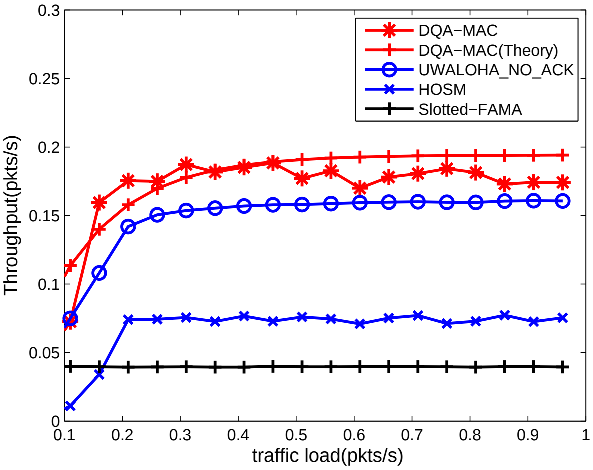
$$Throughput_{theory} = \frac{S \times packet\ rate}{packet\ length}, \tag{27}$$

where $S$ is calculated from (22). It agrees well with the trend of the simulated curve (DQA-MAC). In these figures, we also find an interesting phenomenon, i.e. the network throughput of the DQA-MAC fluctuates along a straight line. The mainly possible reason could be that the period of data transmission phase depends on the waiting transmission time which varies with the traffic loads and the propagation delays.

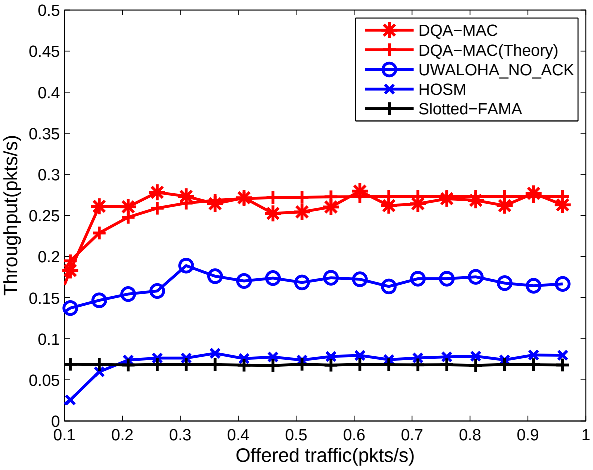
Furthermore, the impact of data packet size on the throughput is verified as well. The packet sizes vary from 50 bytes to 300 bytes with setting $\lambda$ at 1 packets/s. It can be seen in Fig. 5 that as the packet size increases, the throughputs of DQA-MAC, UWALOHA_NO_ACK, and HOSM decrease. This is because, for the DQA-MAC and HOSM, they need to wait more time to transmit data packets. For UWALOHA_NO_ACK, the packets collision increases as the packet size grows. But the throughput of Slotted-FAMA stays unchanged because the slot time does not change with the packet size and packets transmission process remains the same even with different packet sizes.

### 2) End-to-End Delay

Fig. 6 displays the end-to-end delay of these MAC protocols and shows that in general the delay of the protocols increases along with the increase of the traffic loads. The queuing delay, part of the delay, will become larger resulting from the increasing traffic loads. More specifically, the delay of the UWALOHA_NO_ACK is less than the proposed DQA-MAC, because the data packets do not need to wait and can be sent at any wanted time. The DQA-MAC performs better than the Slotted-FAMA because the former only has one way while the latter has two ways to send data packets. However, the HOSM performs better than DQA-MAC indicating that the concurrent transmission of all nodes would require more waiting time.



(a) 6 nodes in the network



(b) 11 nodes in the network

FIGURE 4: Network throughput with varying traffic loads

### 3) Average Energy Consumption

The average energy consumption of these MAC protocols is shown in Fig. 7. In general, the consumption of the protocols decreases to a downbound as the traffic loads grow due to the fact that the energy may run out of when the traffic load gets larger as time goes by. Moreover, the average energy consumption of UWALOHA_NO_ACK is smaller than the proposed DQA-MAC, because the receiver need not send ACK packets to the senders, which saves the energy. The DQA-MAC consumes less energy than Slotted-FAMA since the latter sends two packets in a data transmission round while the former sends only one, and sending packets requires more energy. The Slotted-FAMA exhibits better performance than the HOSM due to one more packet demanded by the HOSM to reserve the nodes transmission time.

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

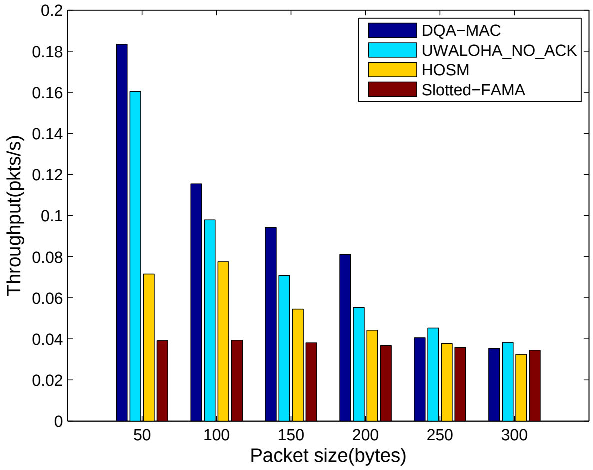


FIGURE 5: Network throughput with different packet sizes

#### 4) Fairness Index

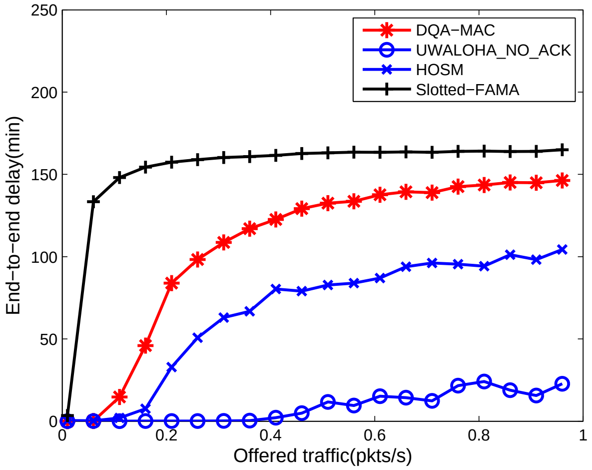
Fig. 7 presents the fairness index of these MAC protocols. The Slotted-FAMA performs the best while the DQA-MAC and HOSM are better than UWALOHA_NO_ACK. This reveals that scheduling based protocols perform better than random access protocols in terms of fairness because the chance of channel access is influenced by distances between senders and receivers in random access protocols. And the smaller the distance is, the bigger the chance of successfully accessing channel is, leading to worse fairness. However, in scheduling based protocols, since the transmission time of nodes is ordered and scheduled regardless of their locations, this results in good fairness performance.

### C. DISCUSSIONS ABOUT MULTI-HOP NETWORKS

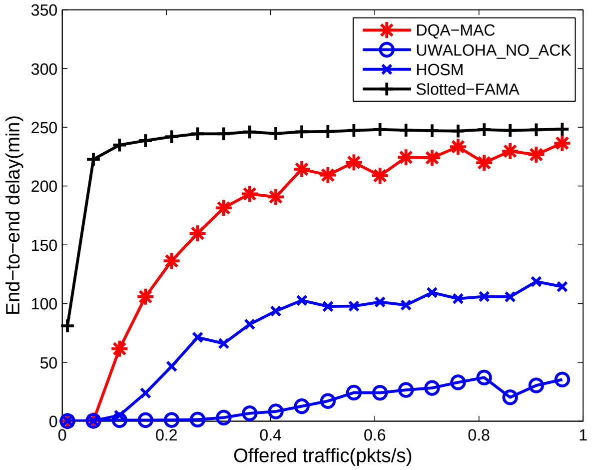
The proposed MAC protocol is mainly designed for the star networks which usually consist of single-hop networks shown in the Fig.1 in this paper. The network coverage area could be expanded by introducing multiple star networks. Besides, for the UASNs, it is also a preferable suggestion to consider a multi-hop network. To apply our proposed MAC protocol into a multi-hop network, a little change is required, i.e. two beacon packets need to be added to switch the process between hops compared to the single-hop network.

In the multi-hop networks, the traffic in the first hop is usually heavier than the other hops. Besides, the access strategies in the hops except the first hop are the same as the second one. So, to simplify the discussion, a two-hop networks is considered. We assume that the sender in the first hop is called SNs, the receiver in the first and second hop is called CN1s and CN2s, respectively. Note that the CN1s also play the role of senders in the second hop.

At first, the initial phase should be carried out both in the first hop and the second hop. The data transmission phase and rescheduling phase of the proposed MAC protocol can be completely applied into the first hop. And then for the second hop, owing to the light traffic, if a CN1 wants to transmit data packets to CN2, it can send a beacon packet to the CN2 and SNs after the end of process in the first hop. When SNs receive the beacon packet from the CN1, they will turn to idle mode and wait to send DATA packet until receiving the next beacon packet. When the CN2 receives the beacon packet, it will start the proposed MAC protocol by sending an ACK packet to CN1s. And the other CN1s can choose to send DATA packets or not based on their own state. This process is the same as our proposed MAC protocol. After that the CN2 sends a short ACK packet to inform the CN1s whether the data packet is received or not. And when CN1s receive the ACK packets, it will send a beacon packet to SNs. This indicates the end of the process in the second hop and triggers the process in the first hop.



(a) 6 nodes in the network

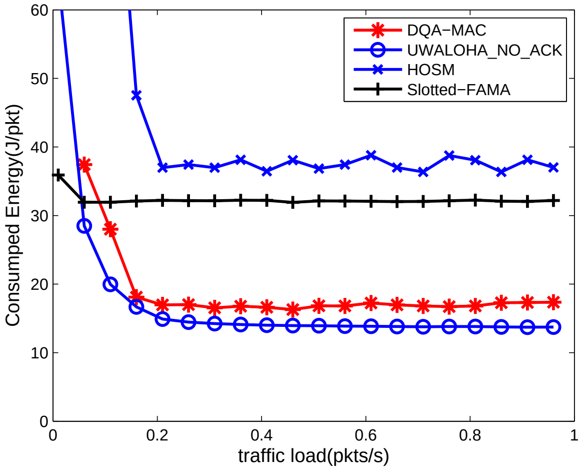


(b) 11 nodes in the network

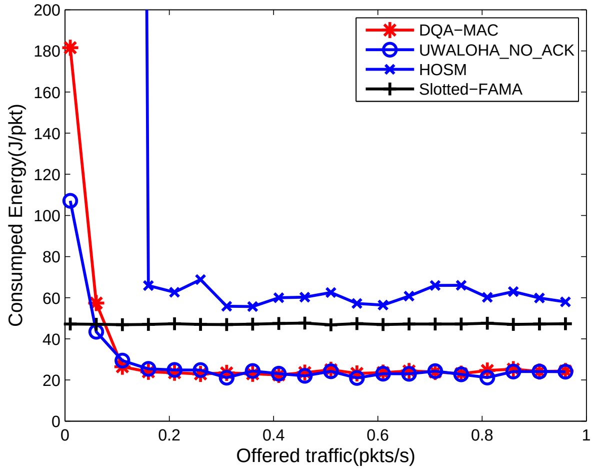
FIGURE 6: End-to-end delay with varying traffic loads

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access



(a) 6 nodes in the network



(b) 11 nodes in the network

FIGURE 7: Average energy consumption with varying traffic loads

## VI. CONCLUSIONS

In this paper, we have designed a DQA-MAC protocol for UASNs with multiple centralized networks. It implements adaptive scheduling transmission, reduction of handshaking packets and concurrent transmission to enhance the performance of the UASNs. The adaptive scheduling transmission improves network throughput and fairness. This protocol reduces the handshaking packets to decrease the end-to-end delay and save energy. The concurrent transmission is carried out to further improve the network throughput and reduce the end-to-end delay. The simulation results demonstrate that compared to the HOSM, UWALOHA_NO_ACK and Slotted-FAMA protocols, DQA-MAC protocol performs well in the network model designed in this paper. However, there are also some limitations in our protocol, such as specific network topology and clock synchronization. The future work will focus on the optimization of the protocol mechanism.

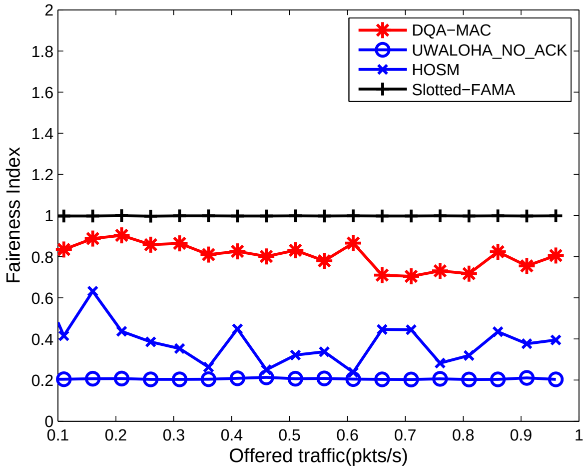


FIGURE 8: Fairness index with varying traffic loads

## REFERENCES

[1] P. Jim, K. Jim and L. B. Neil, "A Survey of Practical Issues in Underwater Networks," ACM SIGMOBILE Mob. Comput. Commun. Rev., vol. 11, no. 4, pp. 23-33, 2007.

[2] J.-H. Cui, J. Kong, M. Gerla and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," IEEE Network, vol. 20, no. 3, pp. 12-18, May. 2006.

[3] Q. Zhao, A. Lambert and C.R. Benson, "FS-MACA: A Future Scheduling MAC Protocol in Underwater Acoustic Networks," In Proc. 9th ACM Int. Conf. on Underwater Networks and Systems, WUWNET '14, New York, USA, 2014, pp. 3:1-3:5.

[4] H. Yu, N. Yao, S. Cai and Q. Han, "Analyzing the performance of Aloha in string multi-hop underwater;acoustic sensor networks," Eurasip Journal on Wireless Commun. and Networking, vol. 1, no. 3, pp. 1-10, Mar. 2013.

[5] C. Petrioli, R. Petroccia and M. Stojanovic, "A comparative performance evaluation of MAC protocols for underwater sensor networks," In Proc. MTS/IEEE OCEANS 2008, Quebec City, Canada, 2008, pp. 1-10.

[6] Y.-j. Chen and H.-L. Wang, "Ordered CSMA: a collision-free MAC protocol for underwater acoustic networks," In Proc. MTS/IEEE OCEANS 2007, Vancouver, Canada, 2007, pp. 1-6.

[7] P. Karn, "MACA: A New Channel Access Method for Packet Radio," In ARRL/CRRL Amateur Radio 9th Comp. Net. Conf., Ontario, Canada, 1990, pp. 134-140.

[8] M. Molins and M. Stojanovic, "Slotted FAMA: a MAC protocol for underwater acoustic networks," In Proc. IEEE OCEANS 2006, Singapore, Singapore, 2007, pp. 1–7.

[9] X. Guo, M. R. Frater and M. J. Ryan, "Design of a propagation-delay-tolerant MAC protocol for underwater acoustic sensor networks," IEEE J. of Ocean. Eng., vol. 34, no. 2, pp. 170-180, Apr. 2009.

[10] H.-H. Ng, W.-S. Soh and M. Motani, "A Bidirectional-Concurrent MAC Protocol With Packet Bursting for Underwater Acoustic Networks," IEEE J. of Ocean. Eng., vol. 38, no. 3, pp. 547-565, Jul. 2009.

[11] S. Shahabudeen, M. Chitre and M. Motani, "MAC protocols that exploit propagation delay in underwater networks," In Proc. MTS/IEEE OCEANS 2011, Waikoloa, HI, USA, 2011, pp. 1-6.

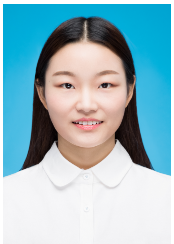
[12] N. Chirdchoo, W.-S. Soh and K. C. Chua, "MACA-MN: A MACA based MAC protocol for underwater acoustic networks with packet train for multiple neighbors," In VTC Spring 2008 - IEEE Veh. Tech. Conf., Singapore, Singapore, 2008, pp. 46–50.

[13] H.-H. Ng, W.-S. Soh and M. Motani, "MACA-U: A media access protocol for underwater acoustic networks," In Proc. IEEE Global Telecommun. Conf. (Globecom), New Orleans, LO, USA, 2008, pp. 323-336.

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access

[14] J. Liu and J. Wang, "A MACA-based collision avoidance MAC protocol for underwater acoustic sensor networks," In Proc. IEEE/OES China Ocean Acoustics (COA), Harbin, China, 2016, pp. 1-4.
[15] N. Chirdchoo, W.-S. Soh and K. C. Chua, "Aloha-based MAC protocols with collision avoidance for underwater acoustic networks," In Proc. IEEE INFOCOM, Barcelona, Spain, 2007, pp. 2271-2275.
[16] C.-C. Hsu, M.-S. Kuo, C.-F. Chou and K. C.-J. Lin, "The elimination of spatial-temporal uncertainty in underwater sensor networks," IEEE/ACM Trans. on Networking, vol. 21, no. 4, pp. 1229-1242, Aug. 2013.
[17] K. Kredo II, P. Djukic and P. Mohapatra, "STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol," In Proc. IEEE INFOCOM, Rio de Janeiro, Brazil, 2009, pp. 2961-2965.
[18] Y. Guan, C.-C. Shen and J. Yackoski, "MAC scheduling for high throughput underwater acoustic networks," In IEEE Wireless Commun. and Networking Conf. (WCNC) 2011, Cancun, Quintana Roo, Mexico, 2011, pp. 197-202.
[19] Y. Noh, U. Lee, S. Han, P. Wang, D. Torres, J. Kim and M. Gerla, "DOTS: A propagation delay-aware opportunistic MAC protocol for mobile underwater networks," IEEE Trans. on Mob. Comput., vol. 13, no. 4, pp. 766-782, Apr. 2014.
[20] Z. Liao, D. Li and J. Chen, "A handshake based ordered scheduling MAC protocol for underwater local area networks," Intern. J. of Distributed Sensor Networks, vol. 11, no. 1, pp. 2-15, Jan. 2015.
[21] N. Chirdchoo, W.-S. Soh and K. C. Chua, "RIPT: A Receiver-Initiated Reservation-Based Protocol for Underwater Acoustic Networks," IEEE J. on Selected Areas in Commun., vol. 26, no. 9, pp. 1744-1753, Dec. 2008.
[22] M. Deng, H. Chen and L. Xie, "DCO-MAC: A Hybrid MAC Protocol for Data Collection in Underwater Acoustic Sensor Networks," Sensors, vol. 18, no. 7, pp. 2300, Jul. 2018.
[23] A.-R. Cho, C. Yun, Y.-K. Lim and Y. Choi, "Asymmetric Propagation Delay-Aware TDMA MAC Protocol for Mobile Underwater Acoustic Sensor Networks," Applied Sciences, vol. 8, no. 6, pp. 962, Jun. 2018.
[24] N. Morozs, P. Mitchell and Y. V. Zakharov, "TDA-MAC: TDMA Without Clock Synchronization in Underwater Acoustic Networks," IEEE Access, vol. 6, pp. 1091-1108, November. 2018.
[25] W.-H. Liao and C.-C. Huang, "SF-MAC: A spatially fair MAC protocol for underwater acoustic sensor networks," IEEE Sensors Journal, vol. 12, no. 6, pp. 1686-1694, Nov. 2011.
[26] M. A. Luque-Nieto, J. M. Moreno-RoldÃąn, J. Poncela and P. Otero, "Reliable transmissions in fair STDMA underwater sensor networks," In 2015 2nd Intern. Conf. on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2015, pp. 1285-1289.
[27] A. Vermeij and A. MunafÃš, "A robust, opportunistic clock synchronization algorithm for ad hoc underwater acoustic networks," IEEE J. of Ocean. Eng., vol. 40, no. 4, pp. 841-852, Oct. 2015.
[28] X. Zhuo, F. Qu, H. Yang and Y. Wu, "Time-Based Adaptive Collision-Avoidance Real-Time MAC Protocol for Underwater Acoustic Sensor Networks," In Proc. of the 13th ACM Intern. Conf. on Underwater Networks and Systems, WUWNET '18, Shenzhen, China, 2018, pp. 1-4.
[29] S. Shakkottai and A. L. Stolyar, "Scheduling for Multiple Flows Sharing a Time-Varying Channel: The Exponential Rule," American Mathematical Society (AMS), vol. 207, pp. 185-201, Dec. 2002.
[30] C. L. Fullmer, J. J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-radio networks," In Proc. of the conf. on Applications, technologies, architectures, and protocols for computer communication (SIGCOMM '95), New York, USA, 1995, pp. 262-273.
[31] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J. Cui, Z. Shi, Y. Fei and S. Zhou, "Aqua-Sim: An NS-2 based simulator for underwater sensor networks," In Proc. of MTS/IEEE OCEANS 2009, Biloxi, MS, USA, 2009, pp. 1-7.
[32] G. A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh and F. Hover, "Communication protocols for underwater data collection using a robotic sensor network," In Proceedings of IEEE GLOBECOM 2011, Houston, TX, USA, 2011, pp. 1–6.

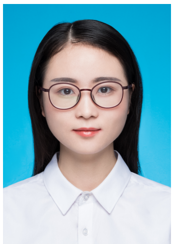


**XIAOXIAO ZHUO** received the B.E. degree in Information Science and Electronic Engineering from Zhejiang University in 2017 and now she is a Ph.D. candidate in the Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Zhejiang University, Zhoushan, China. Her current research interests include media access control protocols and cross-layer design in underwater acoustic sensor networks.

**FENGZHONG QU** received the B.S. and M.S. degrees from Zhejiang University, Hangzhou, China, in 2002 and 2005, respectively, both in electrical engineering. He received the Ph.D. degree in electrical and computer engineering from the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA, in 2009.

From 2009 to 2010, he was an Adjunct Research Scholar in the Department of Electrical and Computer Engineering, University of Florida. Since 2011, he has been with the Ocean College, Zhejiang University, Zhoushan, China, where he is currently a Professor and a Chair of the Institute of Ocean Sensing and Networking. His current research interests include underwater acoustic communication and networks, wireless communications, signal processing, and intelligent transportation systems.

Dr. Qu is an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS and the IET Communications and China Communications.

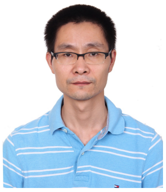


**HONG YANG** is currently working towards the M.S. degree in the Ocean College at Zhejiang University, Zhoushan, China. Her general research interests include MAC protocol design and optimization in underwater acoustic networks.

**YAN WEI** received her B.Sc. an M.Sc. degrees from Wuhan University of Technology, Wuhan, China, in 2004 and 2007, respectively, both in Naval Architecture and Ocean Engineering. She received her Ph.D. degree from the Department of Maritime and Transportation Technology at Delft University, Delft, the Netherlands, in 2012. Since 2012, she has been with the Ocean College at Zhejiang University and currently is an assistant professor. Her current interests include computational fluid mechanics and ocean engineering.

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2913209, IEEE Access



**YEZHOU WU**, received a Bachelor of Science Degree in information and computing science and technology, a master's degree in Applied mathematics, in 2003 and 2006 respectively, both in University of Science and Technology of China. He received his PhD degree in Mathematics in West Virginia University, USA, in 2012. From 2012 to 2014, he was a postdoctor in the Department of Mathematics, the University of Hong Kong. Since 2014, he has been with the Ocean College, Zhejiang University, Zhoushan, China, where he is currently an associate professor. His research interests include combinatorics and graph theory, network topology structure and computer algorithm related research work.

**JIANGHUI LI**, received the B.S. degree in communications engineering from Huazhong University of Science and Technology, Wuhan, China in 2011, the M.Sc. degree in communications engineering, and the Ph.D. degree with receiving the K. M. Stott Prize for excellent research in electronics engineering from the University of York, U.K., in 2013 and 2017, respectively. He has been the first researcher receiving the IEEE OES scholarship in U.K. Since 2017, he has been a Research Fellow with the University of Southampton, U.K. His current research interests include adaptive signal processing, wireless communications, underwater acoustics, and ocean engineering.

• • •

2169-3536 (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.