ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MULTIPLE DATA TRANSMISSION PROTOCOL BASED ON NODE GROUPING IN UNDERWATER SENSOR NETWORKS

[1]WOOJIN KANG, [2]SUNMYENG KIM

[1,2]Department of Computer Software Engineering, Kumoh National Institute of Technology, South Korea

E-mail: [1]shon9811@kumoh.ac.kr, [2]sunmyeng@kumoh.ac.kr

ABSTRACT

Underwater sensor networks (USNs) are characterized by the limited bandwidth and long propagation delay because they use radio waves to transmit data packets. The characteristics pose challenges to the design of medium access control (MAC) protocol. In particular, handshaking based MAC protocols are not appropriate in USNs because of the large delay in exchanging control and data packets. In this paper, we propose a new MAC protocol, called node grouping based multiple data transmission (NG-MDT) protocol to improve performance in USNs. In the proposed protocol, nodes are grouped in a distributed manner according to the distance between themselves and the destination node. When a node obtains a channel access right, it selects one node in the next group. And then, it transmits its own data packet, including the address of the selected node. The selected node also selects another node in its next group, and transmits its data packet including the address of the selected another node. This process is repeated until the last group is reached. In the proposed protocol, multiple nodes transmit their data packets consecutively with one handshaking for channel reservation. Performance evaluation is conducted using simulation, and confirms that the proposed protocol significantly outperforms the previous protocol in terms of throughput, end-to-end delay, and frame success ratio.

Keywords: *Data Transmission, MAC, Node Grouping, Propagation Delay, USN*

1. INTRODUCTION

Underwater sensor networks (USNs) are a class of sensor networks deployed in underwater environments [1]. There are significant differences between UANs and wireless networks because of the unique features such as low available bandwidth and long propagation delay. These features pose challenges to medium access control (MAC) protocol design [2, 3]. And, MAC protocols for wireless networks cannot be directly applied to USNs because they are based on high data rates and negligible propagation delay. Especially, carrier sense multiple access / collision avoidance (CSMA/CA) cannot prevent packet collisions well among nodes due to the long propagation delays in USNs. Therefore, it is necessary to design new MAC protocols to take into account the different features.

Significant efforts have been devoted to the underwater MAC protocol design to overcome the negative effects introduced by the harsh underwater environments [3-7]. Most of them are based on the handshaking in order to reduce the collision probability in USNs. They use control packets such as Request-to-Send (RTS) and Clear-to-Send (CTS) to contend and reserve channel for data transmissions.

Ng, et al. proposed a bidirectional-concurrent MAC (BiC–MAC) protocol based on concurrent, bidirectional data packet exchange to improve the data transmission efficiency [8]. In the BiC–MAC protocol, a sender-receiver node pair is allowed to transmit data packets to each other for every successful handshake. Noh, et al. proposed a delay-aware opportunistic transmission scheduling (DOTS) protocol [9]. In DOTS, each node learns neighboring nodes' propagation delay information and their expected transmission schedules by passively overhearing packet transmissions. And then, it makes transmission scheduling decisions to increase the chances of concurrent transmissions while reducing the likelihood of collisions. In Reference [10], the authors proposed a multiple access collision avoidance protocol for underwater (MACA-U) in which terrestrial MACA protocol was adapted for use in multi-hop UANs. In the MACA-U protocol, a source node transmits a RTS packet to a destination node after channel contention. After receiving the RTS packet, the destination node transmits a CTS packet. And then, the source node transmits its own data packet to the destination node. When other nodes receive the RTS or CTS packets, they set their timer and do not participate in the data packet transmission process. The cascading multi-hop reservation and transmission (CMRT) transmits multiple data packets together with only one handshaking signal to improve channel utilization by [11]. In this way, CMRT can reduce the control packet exchange time and accordingly increase the throughput. The MACA with packet train for Multiple Neighbors (MACA-MN) can overcome the low throughput problem by transmitting a train of packets during each round of handshake [12]. The packet train is formed for multiple neighboring nodes simultaneously.

In the existing handshaking-based MAC protocols, a source node transmits its own data packets to a destination node after getting the channel access right through the handshaking procedure. Therefore, the handshaking protocols cause the low channel utilization due to the presence of long propagation delays in USNs, which in turn severely affects network performance. The CMRT and MACA-MN exhibit short-term unfairness because one node may occupy the channel over short time intervals to transmit multiple data packets.

In this paper, we propose a new MAC protocol, called node grouping based multiple data transmission (NG-MDT) protocol to improve short-term fairness, channel utilization and performance in USNs. In the proposed protocol, nodes are grouped in a distributed manner according to the distance between themselves and the destination node. When a node gets a channel access right, it selects one node in the next group. And then, it transmits its own data packet, which includes the address of the selected node. The selected node also selects another node in its next group, and transmits its data packet including the address of the selected another node. This process is repeated until the last group is reached. In the proposed protocol, multiple nodes transmit their data packets consecutively with one handshaking for channel reservation.

This paper is organized as follows. In section 2, we briefly describe related work. In section 3, the proposed NG-MDT protocol is presented in detail. In section 4, performance studies are carried out through simulation results which is followed by a discussion in section 5. Finally, we draw conclusions in section 6.

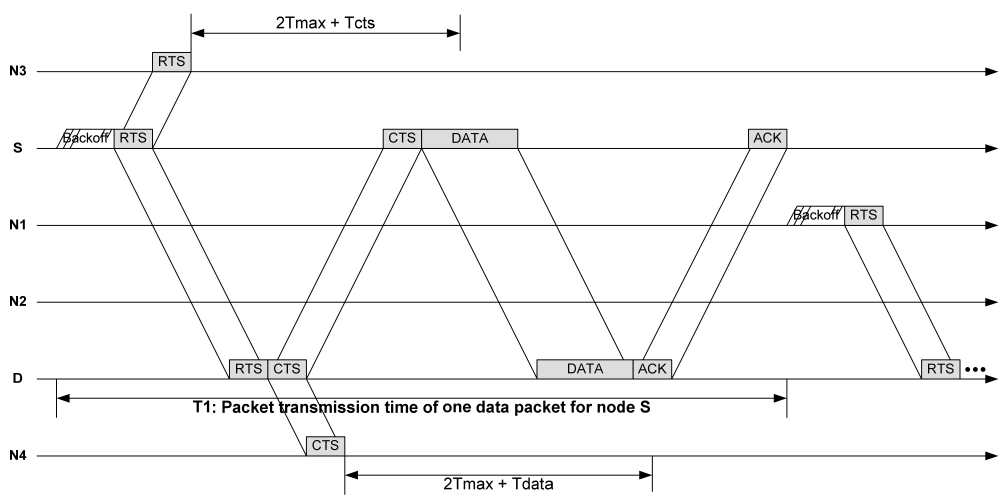

Figure 1: Example Of MACA-U Protocol

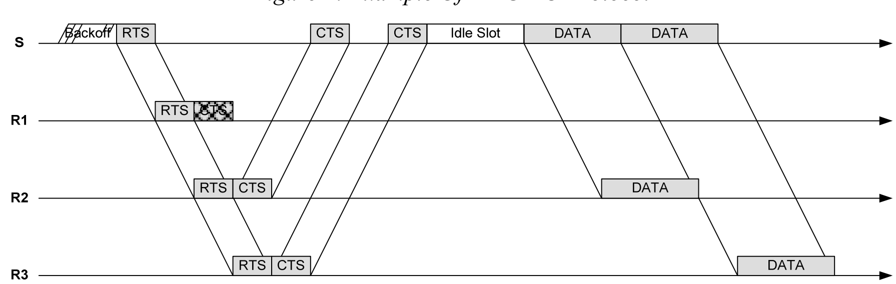

Figure 2: Example Of MACA-MN Protocol

2. RELATED WORK

There have been a large number of studies focusing on MAC protocol in USNs. In this section, the MACA-U and MACA-MN protocols are discussed.

2.1 MACA-U Protocol

In the MACA-U protocol, when a source node has a data packet to send, it uses a handshaking procedure. That is, the source node transmits a RTS packet to a destination node after channel contention. After receiving the RTS packet, the destination node transmits a CTS packet. When other nodes receive the RTS packet, they set their timer to $2Tmax + Tcts$ and do nothing to avoid interference until the timer expires. Similarly, when nodes receives the CTS packet, they set their timer to $2Tmax + Tdata$. $Tmax$ is the maximum propagation delay, $Tcts$ is the CTS duration, and $Tdata$ is the data packet duration. When the CTS packet from the destination node, the source node transmits its own data packet to the destination node. Finally, the destination node sends an ACK packet to the source node.

Figure 1 shows an example of MACA-U protocol. There are six nodes. When the node S has a data packet to send, it starts its backoff procedure. If its backoff counter reaches zero, it sends a RTS packet to the destination node D. The node D sends a CTS packet after receiving the RTS packet. Then the node S transmits its data packet and the node D sends an ACK packet. In order to transmit a data packet for the node S in Figure 1, it takes time of T1. Node N1 also has the same operation as the Node S to transmit its own data packet.

2.2 MACA-MN Protocol

The MACA-MN protocol improves channel utilization by forming a series of packets targeting multiple neighbors during a handshake round, greatly reducing the time wasted due to propagation delays in control packets.

Similar to the widely known MACA protocol, we employ a three way handshake (RTS/CTS/DATA). A source node transmits a RTS packet to a destination node after channel contention. Unlike the MACA protocol, the RTS packets can simultaneously request DATA transmission for multiple receivers. When a receiver hears the RTS packet, it responds with a CTS packet. And then, the source node transmits its own data packets to the receivers. It does not transmit data packets to the receivers from which it does not receive CTS packets correctly. That is, it leaves the requested DATA transmission time slots idle.

Figure 2 shows an example of MACA-MN protocol. There are four nodes: one source node (S) and three receiver nodes (R1, R2, and R3). When the node S has a data packet to send, it starts its backoff procedure. If its backoff counter reaches zero, it sends a RTS packet to the receiver nodes R1, R2, and R3. Both receiver nodes R2 and R3 respond with their CTS packets after receiving the RTS packet. However, the receiver node R1 does not respond with a CTS packet. The node S leaves the first data slot for the receiver R1 idle and transmits its data packets to the receiver nodes R2 and R3.

3. NG-MDT PROTOCOL

3.1 Basic Operation

When a source node has data packets to send, it reserves the channel through RTS/CTS handshaking and then transmit its own data packet to the destination node. After the source node transmits a data packet, neighboring nodes consecutively transmit their data packets following the source node.

Nodes are grouped in a distributed manner according to the distance between themselves and the destination node (we will explain how to group the nodes in sub-section 3.2). When a node in the nth group gets a channel access right, it selects one node in the (n-1)th group (we will explain how to select a node in sub-section 3.3) before transmitting its own data packet. And then, it transmits its own data packet, which includes the address of the selected node. A neighboring node receives the data packet from the source node, and checks whether the packet includes its own address or not. If it includes, the neighboring node becomes another source node, also selects one node in the (n-2)th group, and transmits its data packet including the address of the selected node. Otherwise, it does not do anything to avoid collisions. This process is repeated until the 1st group is reached.

After receiving the data packet from the source node, the destination node checks whether the data packet includes the address of a selected node or not. If it includes an address, the destination node knows that there is another data packet from the selected node. Otherwise, the destination node sends an ACK packet to the source node and the selected nodes.

In the proposed protocol, multiple nodes transmit their data packets consecutively with one handshaking for channel reservation.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

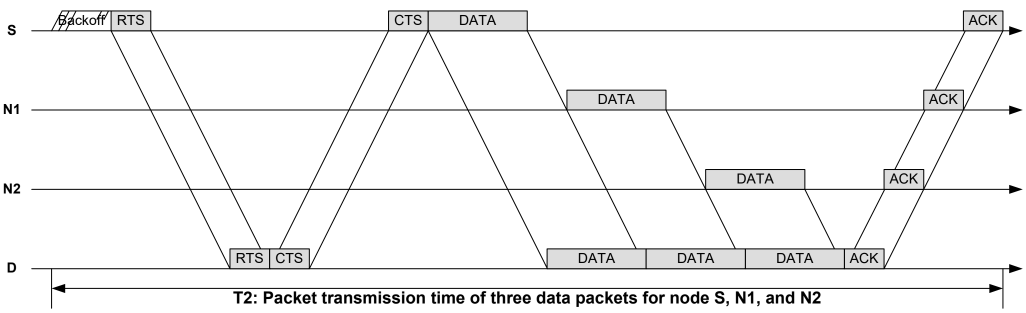

Figure 3: Basic Operation Of NG-MDT Protocol

Figure 3 shows the basic operation of the proposed NG-MDT protocol. There are four nodes. We assume that nodes S, N1, and N2 are in group 3, 2, and 1, respectively. The node S becomes a source node after backoff procedure. And it sends a RTS packet to reserve the channel. The destination node D receives the RTS packet, and then sends CTS packet. After receiving the CTS packet, the node S selects node N1 in the 2nd group, and transmits its own data packet, which includes the address of the node N1. After receiving the data packet from the node S, the node N1 checks whether the data packet includes its own address. If it includes, the node N1 also selects node N2 in the 1st group and transmits its data packet including N2. The destination node D waits for another data packet from the node N1 since the data packet from the node S includes N1 address. The node N2 receives the data packet from the node N1. However, it does not select a neighboring node since its group is the last. Therefore, the node N2 transmits a data packet without including any address. After receiving the data packet from the node N2, the destination node D sends an ACK packet to the nodes S, N1, and N2 because the data packet from the node N2 does not include any address.

In order to transmit three data packets for the nodes S, N1, and N2 in Figure 3, it takes time of T2. In Figure 1, it may take time of 3*T1 to transmit three data packets. This means that the proposed NG-MDT protocol takes less time to transmit data packets compared to the MACA-U protocol.

3.2 Group Decision

Each node determines its own group based on the distance between itself and the destination node in a distributed manner. We can obtain the distance by using the propagation delay and the speed of acoustic signal.

Each node measures propagation delay with its neighbor nodes. Generally, the propagation delay is calculated by using round trip time (RTT). A node updates the propagation delay measurements through the RTS/CTS packet exchange. Figure 4 shows an example of propagation delay measurement. When the node N1 has a data packet to send, it transmits a RTS packet at t_0 to the node N2. After *SIFS* time, the node N2 sends a CTS packet to the node N1. The node N1 finishes receiving the CTS packet at t_1. RTT is the time elapsed from the moment the RTS packet is sent at the node N1 until the CTS packet is successfully received.

$$RTT = t_1 - t_0 \tag{1}$$

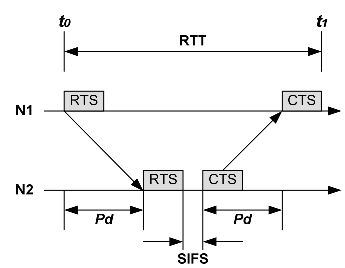

Figure 4: Measurement Of Propagation Delay

The node N1 obtains the propagation delay (*Pd*) as following:

$$Pd = \frac{RTT - RTS - CTS - SIFS}{2} \tag{2}$$

where, *RTS* and *CTS* are transmission times of RTS and CTS packets, respectively.

After measuring the propagation delay, a node converts the propagation delay to the distance (*Dt*) as following:

$$Dt = Pd / v \tag{3}$$

where, v is the sound speed and is 1,500 m/s.

After obtaining the distance to its neighbor node, each node decides its group. We use several terms to decide the group. N is the number of groups.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TRmax is the maximum transmission range. δ is a step size and calculated as (*TRmax* / *N*). A node belongs to the group *k*, if its distance meets the following condition:

$$(k\text{-}1)\cdot\delta < Dt < k\cdot\delta \quad (4)$$

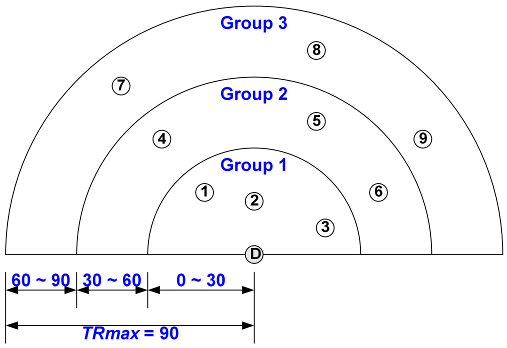

Figure 5: Example Of Group Decision

Node	1	2	3	4	5	6	7	8	9
Distance	25	13	22	35	52	48	67	83	72
Group	1	1	1	2	2	2	3	3	3

Figure 6: Result Of Group Decision

Figure 5 shows an example of deciding a group. We assume *TRmax* is 90, the number of groups is 3. Therefore, the step size is 30 (= 90 / 3). The distances from each node to node D are given as Figure 6. For example, the distance from node 1 to node D is 25. Therefore, the group of node 1 is 1. Figure 6 shows a result of group decision for the nodes in Figure 5.

Address of Neighbor	Propagation Delay	Distance	Group
N_1	$Pd_{(N1)}$	$Dt_{(N1)}$	$G_{(N1)}$
...	...	...	...
N_n	$Pd_{(Nn)}$	$Dt_{(Nn)}$	$G_{(Nn)}$

Figure 7: Group Table

Each node maintains a group table, which has information such as propagation delay and distance between itself and a neighboring node, and group number (see Figure 7).

3.3 Neighboring Node Selection

In this sub-section, we describe how to select a neighbor node.

Each node maintains a neighbor node table, which has information such as distance between a source node and a neighboring node, their addresses and group number (see Fig. 8).

Address of Source	Address of Destination	Distance	Group
S_1	D_1	$Dt_{(S1\text{-}D1)}$	$G_{(S1\text{-}D1)}$
...	...	...	...
S_n	D_n	$Dt_{(Sn\text{-}Dn)}$	$G_{(Sn\text{-}Dn)}$

Figure 8: Neighbor Node Table

To maintain a neighbor node table, each node periodically broadcasts a group (GRP) packet including its own group table to its neighbor nodes. Figure 9 presents a format of a GRP packet. The address field in a GRP packet is specified as a broadcast address, rather than a specific node address, to contact every neighbor node. Source address field is the transmitter of this GRP packet. Number of records means how many information is included. Each record consists of address of destination node, distance, and group in Figure 7.

When a node in the *n*th group gets a channel access right, it selects one of nodes in the (*n*-1)th group before transmitting its own data packet. And then, it transmits its own data packet including the address of the selected node.

A source node in the *n*th group selects a neighboring node in the (*n*-1)th group with the minimum sum of the distance between the source node *S* and a neighboring node *i*, and the distance between the neighboring node *i* and the destination node *D*. Equation (5) shows the neighboring node selection criterion.

$$\min_{i\in M}\{P_{(S-i)}+P_{(i-D)}\} \quad (5)$$

where, *M* is the set of nodes in the (*n*-1)th group.

Frame Control	Broadcast Address	Source Address	Number of Records	Address of Destination	Distance	Group	Address of Destination	Distance	Group	•••	FCS
				1st Record			2nd Record			•••	

Figure 9: Format Of GRP Packet

Frame Control	Duration ID	Address 1	Address 2	Address 3	Sequence Control	Address 4	Address 5	Frame Body	FCS

Figure 10: Format Of Data Packet

After deciding the neighboring node, the source node transmits its own data packet including the address of the selected node. Figure 10 shows the data packet format used in the proposed protocol. Unlike the standard format, we add an address 5 field for the selected node.

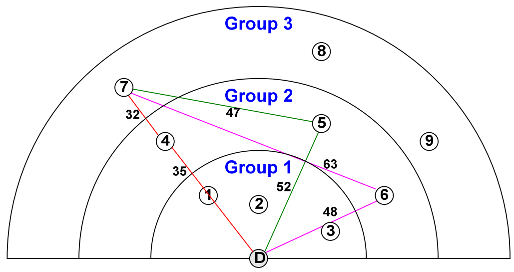

Figure 11: Example Of Selecting A Neighboring Node For Node 7

Figure 11 shows an example of selecting a neighboring node for node 7. The node 7 has channel access right and selects one of nodes 4, 5, and 6 in group 2. The numbers are distances between two neighboring nodes. We assumes that the distances between node 7 and node 4, node 4 and node D, node 7 and node 5, node 5 and node D, node 7 and node 6, node 6 and node D are 32, 35, 47, 52, 63, and 48, respectively. Therefore, the distance summations of nodes 7-4-D, 7-5-D, and 7-6-D are 67, 99, and 111, respectively. Consequently, the node 7 selects the node 4 and enters the address of the selected node in the address 5 field. Finally, it sends the data packet.

When the node 4 receives the data packet from the node 7, it checks whether its address is in the address 5 field or not. It selects one of nodes 1, 2, and 3 in group 1 since its address is included. Figure 12 shows an example of selecting a neighboring node for node 4. The distance summations of nodes 4-1-D, 4-2-D, and 4-3-D are 35, 48, and 71, respectively. Therefore, the node 4 selects the node 1 and enters the address of the selected node in the address 5 field. Finally, it sends the data packet.

When the node 1 receives the data packet from the node 4, it also checks whether its address is in the address 5 field or not. Even though its address is included, it does not select any neighboring node since its group is the last.

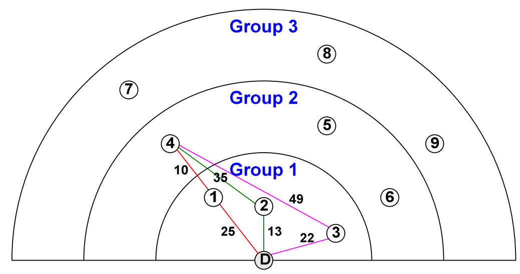

Figure 12: Example Of Selecting A Neighboring Node For Node 4

Figure 13 shows a timing diagram for example of

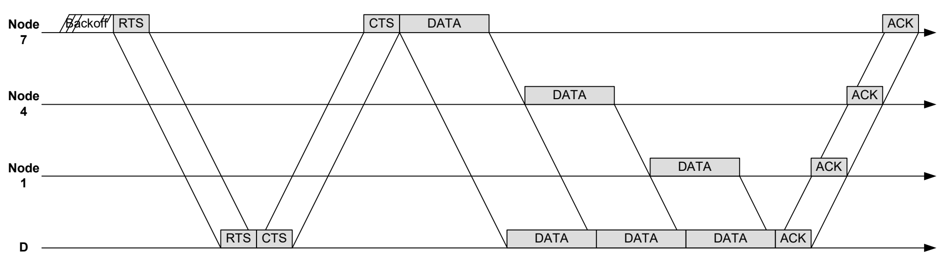

Figure 13: Timing Diagram For Example Of Selecting Neighboring Nodes

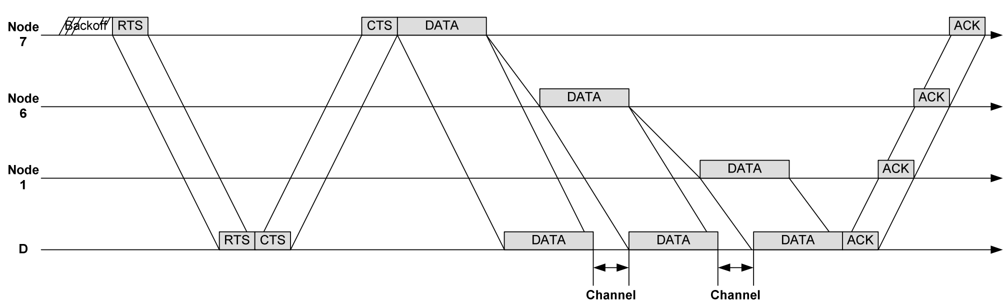

Figure 14: Example Of Channel Time Wastage

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

selecting neighboring nodes.

In the proposed protocol, a source node selects a neighboring node in the next group with the minimum sum of the distances (see the Equation (5)). Instead of this criterion, if we select a neighboring node randomly, then the proposed protocol may waste channel time due to the long propagation delay.

In Figures 11 and 12, if the node 7 selects the node 6 and the node 6 selects the node 1, then it consumes much time to transmit their data packets. Figure 14 shows an example of channel time wastage when a random selection is used. A data packet transmitted by the node 7 arrives late at the distant node 6 due to a long propagation delay. Therefore, the distant node 6 sends its data packet later. Data packets do not arrive at the destination node D consecutively and idle time occurs between the data packets. The node 6 also selects the distant node 1, resulting in idle time at the destination node D. These idle times wastes channel time.

3.4 Backoff Procedure

Backoff procedure is a key component in random access networks. When a node has a data packet to send, it senses the channel to find out whether it is busy or not. If the channel is busy, the node defers its transmission until the end of the ongoing transmission. If the channel is idle for an interval of the distributed interframe space (DIFS), the node starts its backoff timer. When the backoff timer expires, the node starts its transmission.

This backoff time (*BT*) is selected uniformly between 0 and the contention window (*CW*) as following:

$$BT = random(0, CW) \times SlotTime \tag{6}$$

where, *SlotTime* is the duration of a slot time.

The *CW* is an integer between the minimum contention window (*CWmin*) and the maximum contention window (*CWmax*). Each node increases its contention window (*CW*) by *StepSize* if its transmission is collided. When the *CW* reaches *CWmax*, it remains at *CWmax*. It resets its *CW* to *CWmin* if the transmission is successful.

$$CW = \begin{cases} min(CW + StepSize, CW_{\max}), & if\ collided \\ CW_{\min}, & else \end{cases} \tag{7}$$

Figure 15 illustrates the operation of the backoff procedure. There are three nodes in the figure. First, N1 starts its backoff procedure after waiting for the DIFS time interval. When its backoff timer expires, it sends a data packet to N2. N2 replies with an ACK packet after SIFS time interval. N3 has a data packet to send at time t_0. However, it defers its transmission since the channel is busy. When the channel is sensed to be idle, it waits for the DIFS time interval. And then it operates the same procedure as N1.

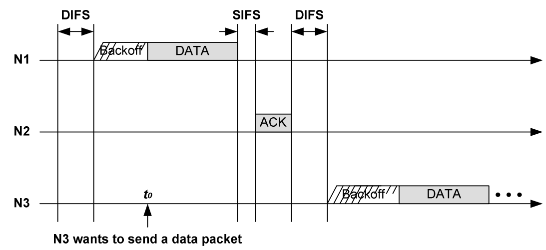

Figure 15: Example Of Backoff Procedure

3.5 Virtual Carrier Sensing

Carrier sensing is carried out in two ways to avoid packet collisions. First, physical carrier sensing detects status (idle or busy) on the channel. Channel is busy when a packet is being transmitted. Second, virtual carrier sensing is based on network allocation vector (NAV) to present a logical status. To implement virtual carrier sensing, the MAC layer packet headers contain a duration field that specifies the amount of time that the medium is to be reserved for transmitting the data packet. After receiving MAC layer packets, nodes obtain the duration field and update their NAV, which is an indicator for a node on how long it must defer from accessing the medium.

Duration fields are set as following rules. After backoff procedure is done, a source node calculates the duration value for an RTS packet (D_{RTS}).

$$\begin{aligned} D_{RTS} = {} & RTS + CTS + DATA + ACK + \\ & SIFS \times 3 + Pd_{\max} \times 4 \end{aligned} \tag{8}$$

where, *RTS*, *CTS*, *DATA*, and *ACK* are transmission times of RTS, CTS, DATA, and ACK packets, respectively. *SIFS* and Pd_{max} are SIFS time and maximum propagation delay.

After receiving the RTS packet from the source node, the destination node calculates the duration value for a CTS packet (D_{CTS}).

$$D_{CTS} = D_{RTS} - RTS - SIFS - Pd_{\max} \tag{9}$$

After receiving the CTS packet from the destination node, the source node selects one of nodes in the next group. If a node is selected, the duration field for a data packet has to include the transmission time of a data packet for the selected node. Otherwise, it does not include the transmission time. We can obtain the duration value

Journal of Theoretical and Applied Information Technology
31st March 2018. Vol.96. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

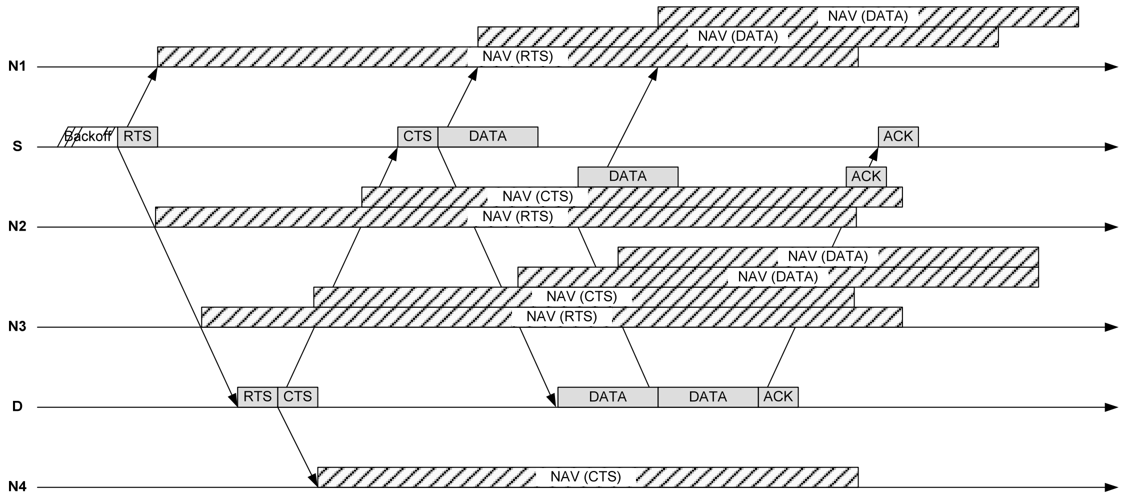

Figure 16: Example Of NAV Updates

for a data packet (D_{DATA}). If a nodes is selected, then the value is as following:

$$D_{DATA} = DATA \times 2 + ACK + SIFS \times 2 + Pd_{\max} \times 2 \quad (10)$$

Otherwise, it is:

$$D_{DATA} = DATA + ACK + SIFS + Pd_{\max} \times 2 \quad (11)$$

Figure 16 shows an example of NAV updates in the proposed protocol. In the figure, there are six nodes. S and D mean source node and destination node, respectively. N1, N2, N3, and N4 are their neighbor nodes. However, N1 and D are outside the transmission range. That is, they are hidden terminals each other. N4 and S are also outside the transmission range. After backoff counter reaches zero, node S sends an RTS packet including the duration field calculated by using equation (8). After receiving the RTS packet, nodes N1, N2 and N3 set their NAVs and do not work to avoid collisions. However, N4 is still working since it is outside the transmission range of node S. Node D calculates the value of duration field by using equation (9) and then sends a CTS packet to node S. Nodes N2 and N3 update their NAV values. Node N4 sets its NAV. After receiving the CTS packet from the node D, node S selects node N2 based on equation (5) and transmits a data packet including the address 5 field of N2 and duration field calculated by using equation (10). After receiving the data packet, nodes N1 and N3 update their NAVs. Node N2 canceled its NAV because the address 5 field in the data packet includes its own address. Node N2 sends its data packet including duration field obtained by using equation (11). After receiving the data packet from node N2, nodes N1 and N3 update their NAVs. Finally, node D sends an ACK packet to node S.

4. SIMULATION RESULTS

In this section, we analyze simulation results of the proposed NG-MDT protocol. To study the performance of the NG-MDT protocol, we actually implemented the protocol in NS3. Performance of the NG-MDT protocol is compared with that of the MACA-U protocol.

Table 1: Simulation Parameters.

Parameter	Value
RTS	40 bits
CTS	40 bits
DATA	256 bits
ACK	40 bits
Slot Time	1500 ms
SIFS	200 ms
DIFS	3200 ms
Data Rate	1500 bps
Sound Speed	1500 m/s
CWmin	10
CWmax	40
Step Size	10
TRmax	1500 m
Number of Groups	3

The system parameters used in the simulation are listed in Table 1. We simulated a with a maximum data rate of 1,500 bps. The length of control packets such as RTS, CTS, and ACK is 40 bits. A constant data packet size of 256 bits was used. Sound speed is 1500m/s.

To generate data packets, we used the saturated traffic model. In this model, queues of every node are always full of data packets.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In the simulation, we consider the topology shown in Figure 17. In the topology, there are several source nodes and one destination node. The source nodes have data packets to send to the destination node. The destination node has no data packets to send and is only a destination node of them.

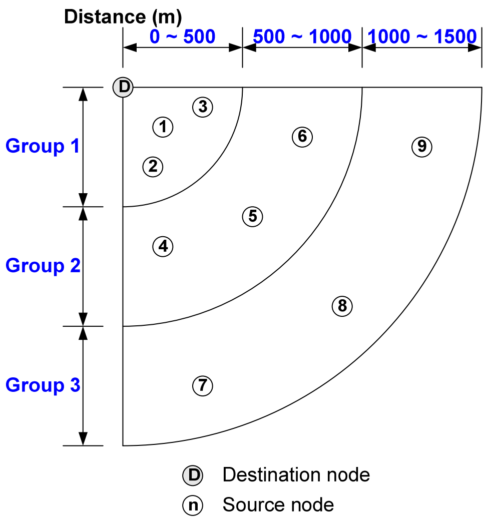

Figure 17: Simulation Topology

The maximum transmission range is set to 1500 m. All source nodes are deployed in a 2-D area of 1500m * 1500m. All source nodes are able to hear each other. The destination node is placed at the point (0, 0). We divide the source nodes into three groups. That is, each group contains 1/3 of the source nodes. Source nodes are uniformly distributed in their group area.

The main performance metrics of interest are throughput, end-to-end delay, and frame success ratio. The end-to-end delay is the time between a data packet arrival at the queue of a source node and the successful data packet transmission to the destination node. The frame success ratio is the number of successful reception frames over the total number of transmitted frames. All simulation results were averaged over 10 simulations.

Figure 18 shows the results for throughput according to the number of source nodes. The proposed NG-MDT protocol always shows better throughput than the MACA-U protocol. The NG-MDT protocol is able to maintain a stable throughput even when the number of source nodes is large. The proposed NG-MDT protocol has about 63% higher throughput than the MACA-U protocol regardless of the number of source nodes.

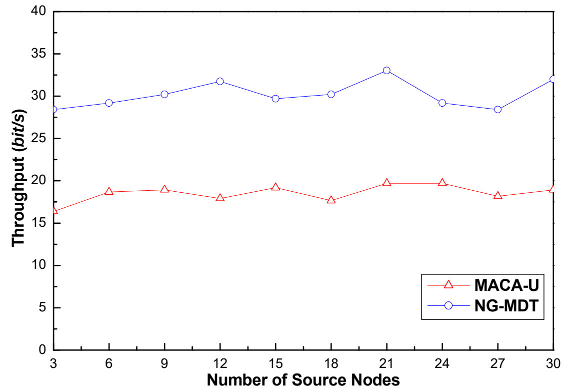

Figure 18: Throughput According To The Number Of Source Nodes

In the proposed protocol, when a source node belonging to group 3 obtains channel access rights, two source nodes can additionally transmit their data packets. In case of a source node belonging to group 2, one source node can additionally transmit a data packet. A source node belonging to group 1 does not select a next source node and provide an additional transmission. Therefore, on average, one additional data packet transmission is possible when one source node gains channel access rights. This means that the proposed protocol can increase the throughput by 100%. However, the throughput in the proposed protocol has increased by 63% compared to the MACA-U protocol as mentioned above. This is due to the propagation delay difference among the source nodes. Source nodes close to the destination node are likely to obtain the channel access right because of the short propagation delay, but far-source nodes have low probability of acquisition because of the long propagation delay. The probability that the source nodes belonging to group 1 acquire channel access rights is increased, and the throughput increase ratio is lowered.

Figure 19 is the results for the end-to-end delay according to the number of source nodes. The NG-MDT protocol always outperforms the MACA-U protocol. Additional source nodes selected from the source nodes belonging to group 2 or group 3 do not perform the backoff procedure. Therefore, it is not necessary to wait for the DIFS time and the backoff interval. There is also no need to transmit RTS and CTS packets, and there is no propagation delay required to transmit these packets. Consequently, the proposed NG-MDT protocol has low end-to-end delay compared to the MACA-U protocol.

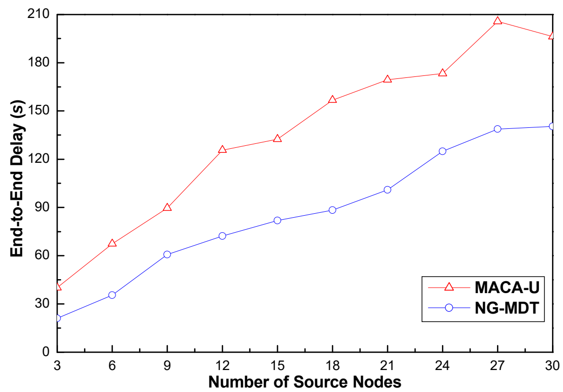

Figure 19: End-to-End Delay According To The Number Of Source Nodes

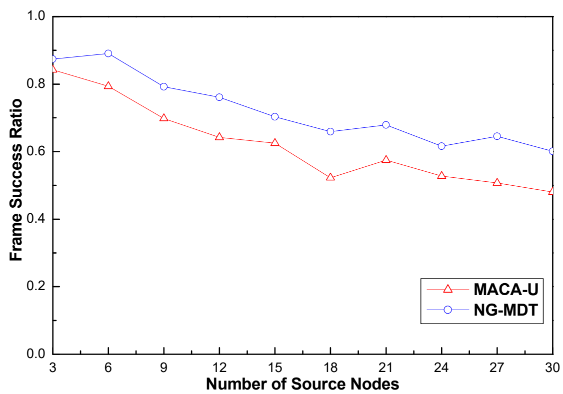

Figure 20: Frame Success Ratio According To The Number Of Source Nodes

Figure 20 depicts the frame success ratio according to the number of source nodes. As the number of source nodes increases, the frame success ratio for the NG-MDT and MACA-U protocols decreases slowly since collision probability increases.

5. DISCUSSION

Significant efforts have been devoted to the underwater MAC protocol design to overcome the negative effects introduced by the harsh underwater environments. Most of MAC protocols for USNs focus on the contention-based techniques since they facilitate an easy deployment on nodes. They use control packets to contend and reserve channel for data transmissions. Contention-based MAC protocols have low channel utilization and performance due to the large delay in exchanging control and data packets.

The previous MACA-MN protocol improves channel utilization by forming a series of packets targeting multiple receivers during a handshake process. Default deployments of USNs are based on a many-to-one communication where a single sink collects data from multiple sensors. The many-to-many communication with multiple receivers that the MACA-MN protocol considers are not common.

The main contributions of the proposed NG-MDT protocol are summarized as follows: 1) The proposed protocol improves channel utilization and performance of USNs in a default many-to-one communication environment. 2) Nodes are grouped in a distributed manner according to the distance. When a node gets a channel access right, it helps another node in the next group to send data packets. Therefore, it increases the network throughput and reduces the end-to-end delay. 3) The proposed protocol presented an efficient neighboring node selection criterion. Therefore, it minimizes the channel time waste that can be caused by selecting a neighboring node in the next group. 4) We implemented the proposed protocol based on NS3, and demonstrated that our protocol is correct and efficient.

6. CONCLUSIONS

USNs use handshaking based MAC protocols. They are not appropriate in USNs because of the large propagation delay in exchanging control and data packets. In this paper, we proposed the NG-MDT protocol to improve performance in USNs. In the proposed protocol, source nodes are divided in several groups based on the distance to the destination node. When a source node gets a channel access right, it selects one node in the next group. And then, it transmits its own data packet including the address of the selected node. The selected node transmits its data packet following the data transmission of the source node. In the proposed protocol, multiple nodes transmit their data packets consecutively with one handshaking for channel reservation. Simulation results show that the proposed protocol works well and improves network performance.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1A01059644).

REFRENCES:

[1] P. Casari and M. Zorzi, "Protocol Design Issues in Underwater Acoustic Networks", *Computer Communications*, Vol. 34, No. 17, Nov. 2011, pp. 2013-2025.

[2] Y. Zhu, Z. Peng, J.-H. Cui, and H. Chen, "Toward Practical MAC Design for Underwater Acoustic Networks", *IEEE Transactions on*

Mobile Computing, Vol. 14, No. 4, April 2015, pp. 872-886.

[3] L. Pu, Y. Luo, H. Mo, S. Le, Z. Peng, J.-H. Cui, and Z. Jiang, "Comparing Underwater MAC Protocols in Real Sea Experiments," *Computer Communications*, Vol. 56, Feb. 2015, pp. 47-59.

[4] Y. Zhang, Y. Chen, S. Zhou, X. Xu, X. Shen, and H. Wang, "Dynamic Node Cooperation in an Underwater Data Collection Network", *IEEE Sensors Journal*, Vol. 16, No. 11, 2016, pp. 4127-4136.

[5] 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 Transactions on Networking*, Vol. 21, No. 4, 2013, pp. 1229-122.

[6] W. H. Liao, C. C. Huang, "SF-MAC: A Spatially Fair MAC Protocol for Underwater Acoustic Sensor Networks", *IEEE Sensors Journal*, Vol. 12, No. 6, 2012, pp. 1686-1694.

[7] S.-M. Xiong, C. Yuan, L. Tian, and Y. Zhan, "RET-MAC: A New Fair MAC Protocol for Underwater Acoustic Sensor Network", *International Journal of Distributed Sensor Networks*, Vol. 9, No. 6, 2013.

[8] H.-H. Ng, W.-S. Soh, and M. Motani, "A Bidirectional-Concurrent MAC Protocol with Packet Bursting for Underwater Acoustic Networks", *IEEE Journal of Oceanic Engineering*, Vol. 38, No. 3, July 2013, pp. 547-565.

[9] 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 Transactions on Mobile Computing*, Vol. 13, No. 4, April 2014, pp. 766-782.

[10] H.-H. Ng, W.-S. Soh, and M. Motani, "MACA-U: A Media Access Protocol for Underwater Acoustic Networks", *IEEE Globecom 2008*, Nov. 2008, pp. 1-5.

[11] J.-W. Lee and H.-S. Cho, "Cascading Multi-hop Reservation and Transmission in Underwater Acoustic Sensor Networks", *Sensors*, Vol. 14, 2014, pp. 18390-18409.

[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", *Proceedings of IEEE Vehicular Technology Conference*, 2008.