I. INTRODUCTION

Underwater acoustic networks (UANets) have opened new research opportunities and promise great potential for underwater exploration, resource monitoring, and military applications [1]. Simultaneously, we have faced remarkable limitations of UANets compared with terrestrial wireless networks (TWNets), including narrow bandwidth, a low data rate, and a large propagation delay [2-4]. Under these restricted network environments, a medium access control (MAC) protocol can play a crucial role in enhancing overall performance by efficiently managing the channel access of underwater nodes (UNs) [5]. A large volume of literature on MAC protocols for UANets has been produced [6, 7]. Designing a proper MAC protocol for a UANet executing a target application is still in progress [8, 9].

Without loss of generality, a MAC protocol for UANets is required to mitigate their aforementioned intrinsic restrictions. It is remarkable that the propagation delay of acoustic waves in underwater is five orders of magnitude higher than that of radio waves in the air [10]. The transmission delay underwater is also higher than that in the air due to the slow data rate of underwater acoustic channels [11]. Under these fixed delays determined by the physical layer, a MAC protocol is required to reduce overall end-to-end delay by efficiently managing the channel access of UNs.
It is also necessary for a MAC protocol to consider the characteristics of UANet applications. There are a variety of UANet applications, such as scientific monitoring or surveillance underwater [12]. These UANet applications are mostly on-demand, which implies that UNs are deployed and requested to work only once demand arises. In these UANet applications, UNs (e.g., autonomous underwater vehicles [AUVs], acoustic sensor motes, and submarines) move around underwater at different speeds. Accordingly, a MAC protocol needs to manage various on-demand UANet applications, as well as consider UNs’ mobility.

The overall topology is another consideration upon designing a MAC protocol. In practice, the data obtained from several UANet applications are mainly sent to a terrestrial base station, which can be connected to IP-based backbone networks. Namely, from a network perspective, the overall data from UNs to a terrestrial base station are passed through a hierarchical network architecture consisting of a TWNet and a UANet. In a TWNet, multiple surface stations, which gather data from several UNs and forward the data to a terrestrial base station, are deployed on the water’s surface. A UANet is composed of several clusters. The communication range of a cluster is determined as the one-hop transmission distance between a surface station and UNs [13, 14]. Under this hierarchical network structure, multiple access of surface stations can be mutually related to that of UNs since important information (i.e., UNs’ mobility information and the number of UNs in a cluster) can directly affect the ingress traffic load flown into each surface station. Accordingly, it is expected that a hybrid MAC protocol considering this hierarchical network structure could be more appropriate than two separate MAC protocols, respectively, corresponding to the two networks.

In the literature, several hybrid MAC protocols have been proposed for hierarchical networks. Commonly in the literature, MAC protocols among surface stations are confined to time division multiple access (TDMA), but those among UNs vary, and include TDMA, code division multiple access (CDMA), and even a contention-based MAC protocol. In [14], a MAC protocol consisting of TDMA and CDMA does not consider the mobility of UNs such that idle time slots may occur when the UNs move around from cluster to cluster. In [12], the MAC protocol consists of TDMA and a distributed coordination function. Although the MAC protocol allocates a fixed number of time slots into UNs according to the mobility of the UNs, several UNs that are not allowed to use a time slot are still idle, which increases the overall delay. In [15], another MAC protocol consists of two TDMA scheduling, where a portion of the UNs are only pre-allocated to use the time slots without considering the mobility of UNs, such that the remaining UNs are still idle.

In this paper, we propose a new MAC protocol, referred to as hierarchical time division multiple access (HTDMA). Although TDMA may not suit UNs’ mobility because of the high setup cost and difficulty of frequent schedule changes, HTDMA is based on TDMA because of the advantages of TDMA, including its simplicity, collision-free nature, energy-efficiency, and lack of control signaling [12]. To consider frequent changes in the mobility information of UNs, we make surface stations share the mobility information in a round-robin manner and broadcast the information to their UNs. Thus, HTDMA can satisfy the above considerations and utilizes the mobility of UNs to reduce the overall delay. HTDMA is a hybrid MAC protocol consisting of two TDMA scheduling; TDMA1 for a TWNet and TDMA2 for a UANet. TDMA1 periodically pre-assigns all of the surface stations to a fixed time slot as normal TDMA does. Each surface station forwards data received from diverse UANet applications to a terrestrial base station, and it also broadcasts mobility information of UNs to other surface stations in its pre-allocated time slot. Although contention-based MAC protocols can also be good candidates, TDMA1 is considered for a TWNet due to the advantages of TDMA, as explained earlier. TDMA2 is on-demand; several TDMA2s can be concurrently executed under TDMA1 with respect to ongoing applications. In one application, TDMA2 is implemented in each cluster, respectively, and the corresponding time slot allocation is instantaneously determined by the mobility information of the UNs provided by the surface station. By doing this, HTDMA can overcome the intrinsic limitations of TDMA such as its weakness in dealing with mobility and the occurrence of idle time slots. Accordingly, it is expected that HTDMA may reduce end-to-end delay, as well as improve channel utilization by accommodating various UANet applications.

This paper is structured as follows. We outline previous works on MAC protocols targeted for UANets in Section II. A UANet topology is described in Section III. In Section IV, we detail HTDMA. In Section V, the delay of HTDMA is analytically investigated and compared with those of TDMA and Slotted-ALOHA. Finally, we conclude this paper in Section VI.

II. PREVIOUS WORKS ON MAC PROTOCOLS FOR UANETS

In this section, we give a brief summary of previous works on MAC protocols designed for UANets. These MAC protocols are broadly categorized into two types: contention-based and contention-free MAC protocols [13]. Contention-free MAC protocols can be advantageous over contention-based MAC protocols in that they can avoid the
message overhead in protocols such as request-to-send and clear-to-send handshaking of carrier sense multiple access (CSMA) or multiple access with collision avoidance (MACA) [16]. In addition, contention-free MAC protocols are delay-tolerant while contention-based MAC protocols such as ALOHA or Slotted-ALOHA result in severe delays in heavy traffic-loaded environments [17]. Even CSMA and MACA may result in significant random back-off delay when the overall network traffic load increases. Contention-based MAC protocols also induce the hidden- or exposed-terminal problem.

TDMA, frequency division multiple access (FDMA), and CDMA are contention-free MAC protocols. Among them, FDMA is considered inefficient for UANets due to the lack of acoustic frequencies that originate from limited bandwidth of the acoustic channel [18]. CDMA can reduce contention and has other benefits for UANets. CDMA requires code assignment and a power control algorithm in order to reduce contention among UNs and a near-far problem. This results in high hardware complexity, decreasing already limited data rates, and thus wasting more energy [1, 17]. TDMA, a simple schedule-based approach with a single frequency, can save energy by allowing UNs to turn on their systems only during scheduled time slots without wasting energy due to idle listening and collision [15]. TDMA has recently attracted significant attention for many UANet applications [19-21]. In spite of its advantages, TDMA is still weak in dealing with the mobility of UNs, which is a significant requirement of current UANet applications [13].

TDMA is also extended to a hybrid MAC protocol in order to be suitable for a clustered network topology. The proposed hybrid MAC protocols commonly consist of an intra-cluster MAC protocol and an inter-cluster one. An intra-cluster MAC protocol corresponds to multiple access of several UNs in a cluster. An inter-cluster MAC protocol manages multiple access of several surface stations. Commonly in the literature, the intra-cluster MAC protocol is confined to TDMA, but inter-cluster MAC protocols have included TDMA, CDMA, and even a contention-based MAC protocol. In [14], the authors propose a MAC protocol consisting of TDMA and CDMA. Each cluster with a specific code supports a fixed number of time slots using TDMA. The authors focus on assigning a limited number of codes without overlapping in an inter-cluster MAC protocol. However, this MAC protocol consequently does not consider the mobility of UNs, such that idle time slots may occur when UNs move around from cluster to cluster. In [12], the authors propose a MAC protocol that consists of TDMA and a distributed coordination function. According to the network topology and mobility of UNs, they allocate a fixed number of time slots to the UNs and surface stations in TDMA. Although they can minimize the number of idle time slots, some UNs, not allowed to use a time slot, can still become idle such that the overall delay may increase. In [15], a MAC protocol combining two TDMA scheduling is introduced. As in [12], a portion of the UNs are only pre-allocated to use the time slots without considering the mobility of the UNs, which is statistically determined such that the remaining UNs are still idle.
II. NETWORK ARCHITECTURE DESCRIPTION FOR UANET APPLICATIONS

In this section, we describe an overall network topology executing several applications for UANets concurrently. Target applications can be specified as on-demand applications for the short-term and wide-area exploration and detection. The practical use of such applications is exemplified by underwater detection using mobile UNs like AUVs, cooperative tasks using mobile UNs like underwater robots in a wide area, underwater exploration (e.g., oil fields or reservoirs), assisted navigation (e.g., identifying hazards or submerged wrecks on the seabed), or mine reconnaissance [22]. As illustrated in Fig. 1, the network consists of two sub-networks, including a TWNet and a UANet.

In the TWNet, several surface stations are deployed on the surface of the water. A surface station has a significant role: connecting the UNs to a terrestrial base station. To do this, a surface station has two communication modes (i.e., terrestrial wireless communication and underwater acoustic communication). That is, a surface station gathers data sent by UNs currently residing in its communication territory using underwater acoustic communication, and also stores the data in its buffer, converting the data into a proper packet format, and forwards the data to a terrestrial base station using terrestrial wireless communication. In addition, a terrestrial base station can send control packets to all surface stations by using a different RF frequency, which informs them of the installation of a new UANet application. Like the UNs, when surface stations experience channel contention with other surface stations medium access among surface stations is also considered in the data-link layer.

A UANet contains multiple clusters, which are respectively governed by a corresponding surface station. Once a specific UANet application is requested, multiple UNs, which are equipped with the same communication systems and sensors, are deployed in an overall UANet: each cluster momentarily has several UNs. As the moving distance can be longer than the communication range of a surface station, the deployed UNs move from one cluster to another: the moving speed of AUVs is around a few meters per second. In addition, the distance between the clusters is less than one hundred meters. When a UN is released, it tries to determine the nearest cluster by sensing the signaling power of a surface station (e.g., BEACON packet). After the UN receives the BEACON packet, it tries to register with the surface station via authentication and then send data, if permitted. When the UN enters a new communication territory of another surface station, its channel access and data transmission is instantaneously managed by a new surface station. As far as mobility is concerned, the mobility information of the UN may be shared among all of the surface stations via handover among surface stations. These activities are totally dependent on HTDMA, a MAC protocol presently in use, which are explained in Section IV.

IV. HTDMA MAC PROTOCOL

HTDMA consists of scheduling of two TDMA protocols: TDMA1 and TDMA2. TDMA1 is constantly executed among surface stations, but TDMA2 is executed when a specific UANet application is requested. In TDMA1, a surface station gathers data from UNs currently residing in its cluster. The surface station delivers the data to a terrestrial base station in its assigned time slot in TDMA1. The surface station also obtains the mobility information of the UN which is currently in its cluster and send its mobility schedule (i.e., leave or stay in a current cluster). After receiving the mobility information of all of the UNs belonging to its cluster, the surface station shares the mobility information with other surface stations in TDMA1. Collecting mobility information between a surface station and a UN and sharing the information among surface stations can be costly. As a UN sends data and mobility information to a surface station at the same time, the size of the packets can increase. Similarly, the size of a surface station’s packets can also increase. At the expense of the mobility information, surface stations can determine the frame size of TDMA2 on the basis of the mobility information in TDMA1. In TDMA2, a surface station assigns time slots to UNs that are currently in its cluster by broadcasting a BEACON packet at the start of one TDMA2 frame. Although a UN may take energy by listening to BEACON packets, its battery can be recharged when they return. Thus, we do not consider energy-efficiency upon
designing HTDMA. The UNs transmit data packets to their surface station in its pre-allocated time slot. We define the network parameters as shown in Table 1 and consider the following premises upon employing HTDMA.

1) Due to remarkably longer propagation and transmission delays in the acoustic channel, the length of one TDMA2 time slot ($T_{S2}$) can be much longer than that of one TDMA1 time slot ($T_{S1}$). We determine $T_{S2}$ as $M \times T_{S1}$. The case of $M = 1$ (i.e., only one surface station in one cluster) is not considered because a surface station should exchange the mobility information of its UNs with other surface stations.

2) $T_{S2}$ is only used toward a UN for a UANet application. $T_{S1}$ is also employed toward a surface station but for multiple on-going UANet applications.

3) $T_{S2}$ consists of the maximum propagation delay in a cluster, transmission delay, and guard time required for time synchronization.

4) The length of the BEACON time slot and that of the time slot pre-allocated to a UN is assumed to be equivalent for simplicity. However, the contention period (CP) time slot reserved for new UNs should be flexible according to the mobility of the UNs. If the mobility is high, the length of the CP time slot should increase.

5) As HTDMA is a TDMA-based MAC protocol, time synchronization is strongly required to synchronize the start and end time points for a time slot underwater. We assume that HTDMA is executed by accompanying a proper time synchronization method such as [23, 24].

Specifically, TDMA2 is initiated, operated, and terminated through three stages; the initialization, normal operation, and termination stages. In the initialization stage, each surface station determines the number of time slots assigned to UNs initially residing in its communication territory once TDMA2 is requested as a response to a new UANet application. In the normal operation stage, a mobile UN sends data packets to a current surface station. The frame size of TDMA2 varies according to the overall mobility pattern of the UNs. The termination stage is requested by a surface station or a terrestrial base station when an on-going UANet application ends. Below, we further detail HTDMA according to three stages.

### A. Initialization Stage

The initialization stage basically starts when a specific UANet application is requested. The initialization stage is executed step by step as follows.

1) A terrestrial base station preliminarily sends several pieces of information, including the number of deployed UNs ($N$) and UNs’ MAC addresses, and initial identification numbers (i.e., 1 to $N$) to all surface stations when a new UANet application is ready.

2) Each surface station consents to start TDMA2 targeted for a requested application during one TDMA1 frame, as shown in Fig. 2.

3) At the time point after one more TDMA1 frame passes, the TDMA2 initialization frame starts, which consists of one BEACON time slot and $N$ time slots, as illustrated in Fig. 2. In the BEACON time slot, a surface station sends time slot allocation information to multiple UNs in its cluster. Then, those UNs inform the surface station of their presence in their pre-assigned time slot, respectively. Each surface station takes as

![Fig. 2. An illustration of time slot allocation of the initialization stage. TDMA: time division multiple access, UN: underwater node, UANet: underwater acoustic network.](image-url)
much as $M \times (N + 1) \times T S_1 + M \times T S_1$ to executed the TDMA2 initialization, where $TS_2 = M \times TS_1$. During this TDMA2 initialization frame, a surface station receives presence information on its UNs and also continuously broadcasts this information to other surface stations via its TDMA1 time slot.

4) After the TDMA2 initialization frame, a surface station $i$ determines the number of time slots of TDMA2, $n_i$, corresponding to UNs presently residing in its cluster. Then, the surface station broadcasts $n_i$ information to the other surface stations.

5) When all of the surface stations have broadcast $n_i$ information, they each send a BEACON packet to their respective UNs in a UANet using TDMA2. This is the start of TDMA2 targeted for a specific UANet application.

6) Even during an on-going UANet application, the initialization stage is also requested by a surface station when it detects the failure of time synchronization.

**B. Normal Operation Stage**

The normal operation stage continues until an ongoing UANet application is terminated. A surface station maintains the mobility status of UNs presently located in its cluster and shares the information with other surface stations using TDMA1. When TDMA2 starts, a surface station $i$ allocates a new identification number $j$, only available in $i$’s cluster, into its UNs by sending a BEACON packet. This procedure is directly related to TDMA2 time slot allocation. Namely, $(i, j)$ implies that a UN in a cluster $i$ can use the $j$-th time slot.

By receiving the BEACON packet, a UN can send data packets to the surface station during its pre-allocated time slot. At the same time, the UN is required to send its mobility schedule (i.e., stay or leave) to the surface station, which broadcasts both data and mobility information to other surface stations in its TDMA1 time slot. Due to the existence of mobile UNs that enter or leave a cluster, a surface station $i$ calculates $n_i$ and allocates time slots into UNs frame by frame by using obtained mobility information. Let us denote a UN that is presently pre-assigned at the $j$-th time slot in a cluster $i$. If UNs, which are pre-assigned earlier than the UN, leave the cluster, the UN can be pre-allocated earlier than the $j$-th time slot in the next TDMA2 frame.

A UN can know whether it enters a new cluster by receiving a BEACON packet, which specifies the identification of a surface station and the starting time point of the next CP time slot. Thus, the UN can try to notify a new surface station of its arrival in the CP time slot. Then, the surface station allocates the UN to the last time slot in the next TDMA2 frame by sending a new BEACON packet, as specified in Fig. 3. After receiving the BEACON packet, the UN can send its data and mobility information during its time slot in the next TDMA2 frame.

**C. Termination Stage**

The termination stage occurs when an ongoing UANet application ends. The termination stage is also executed as follows.

1) A terrestrial base station informs all surface stations of stopping a pending UANet application.

2) After an ongoing TDMA2 frame corresponding to the application is over, a surface station sends a BEACON packet to its UNs, implying the end of TDMA2, as illustrated in Fig. 4.

3) When a UN receives the BEACON packet, it stops sending data packets any longer and prepares to return.

4) The surface station informs the other surface stations of the end of TDMA2 targeted for the ongoing UANet application.
5) When all of the surface stations have broadcasted their TDMA2 termination, they inform the terrestrial base station of the termination in their TDMA1 time slot.

In the termination stage, more than one UANet application can be finished by following the above termination procedures.

V. PERFORMANCE ANALYSIS OF HTDMA MAC PROTOCOL

Decreasing the delay for UANets has been a crucial issue in the literature because the propagation delay of acoustic signals is five orders of magnitude higher than that of RF signals [25–28]. In this section, we analytically investigate the delay of HTDMA. We also derive the delays of TDMA and Slotted-ALOHA and compare them with that of HTDMA. Before the derivation of delay, the mobility of a UN is primarily modeled, as illustrated in the following subsection.

A. UN Mobility Modeling

Let us denote $\alpha_i(s)$ as the number of ingress and egress UNs to a cluster $i$ at the end of one TDMA2 time slot $s$ where $s \geq 1$. The TDMA2 time slot starts to count at $s = 1$. We also refer to $n_i(s)$ as the number of UNs at a cluster $i$ at the end of time slot $s$. It is assumed that the initial condition of $\alpha_i(s)$ is given as $\alpha_i(1) = 0$ at $s = 1$. Assuming that there are initially the same number of UNs in all clusters at $s = 1$, $n_i(1)$ can be obtained as $M$.

At an arbitrary TDMA2 time slot $s$ in a cluster $i$, $\alpha_i(s)$ varies in random order because UNs move cluster to cluster in random fashion. Hence, the set of $\alpha_i(s), \alpha_{i-1}(s), \ldots, \alpha_{i-M}(s), \alpha_{i+1}(s)$, is a $(1 \times M)$ random vector. In addition, the number of UNs in a cluster $i$ at time slot $s$, $n_i(s)$ totally depends on the sum of the present and previous $\alpha_i$ quantities (i.e., $\alpha_i(s), \alpha_i(s-1), \ldots, \alpha_i(2), \alpha_i(1)$). Accordingly, $n_i(s)$ can be expressed as

$$n_i(s) = \begin{cases} n_i(1), s = 1 \\ n_i(1) + \sum_{l=1}^{s} \alpha_i(l), s \geq 2. \end{cases}$$

When a UN moves from one cluster to another, it cannot receive any BEACON packets such that it cannot be registered by any surface stations. This implies $\sum_{i=1}^{M} \alpha_i(s) \leq 0$ and $\sum_{i=1}^{M} n_i(s) \leq N$.

B. Delay Parameter Derivation

1) The delay of HTDMA

Let us denote the TDMA2 frame index as $k(k \geq 1)$, as shown in Fig. 5. While $s$ corresponds to one TDMA2 time slot, $k$ corresponds to one TDMA2 frame. As we modeled the mobility of UNs, the number of UNs, which instantaneously belongs to a cluster $i$ at TDMA2 time slot $s$, $n_i(s)$, may vary as $s$ increases. In a similar manner, we can also consider a new parameter $n_{ik}$, referred to as the number of UNs staying in a cluster $i$ at the $k_{th}$ TDMA2 frame. $n_{ik}$ varies as $k$ increases but is fixed during the $k{th}$ TDMA2 frame. We determine $n_{ik}$ as $n_i(s)$ at the CP time slot of the $(k-1){th}$ TDMA2 frame because the final value of $n_i(s)$ in the previous TDMA2 frame should to be considered.
We denote the probability that the UN stays in a cluster illustrated in Fig. 5, as $P_{\text{DHiC}}(k)$. The UN can probabilistically stay at a kth frame, the delay of the UN using HTDMA at the corresponding delay as $\Delta d_{\text{HTDMA}}(j)$. We also denote the probability that the UN moves to another cluster $i_i$ (case 1) or move to another cluster $i_{ii}$ (case 2).

The delay is defined as the time in which a UN waits over, which is a fixed duration because $\Delta d_{\text{HTDMA}}(j)$ is derived as follows. As shown in Fig. 6, $d_{\text{HTDMA}}(j)$ is determined as $(n_{i} - j + 1) \cdot TS_{2}$. $\Delta d_{\text{HTDMA}}(j)$ depends on the mobility status of $(j - 1)$ preceding the UNs. We denote $h$ as the number of reducible TDMA2 time slots. According to the mobility status of the preceding UNs, $h$ varies, as illustrated in Table 2. For example, if all of the preceding UNs stay in a cluster $j$, $h$ is zero and $\Delta d_{\text{HTDMA}}(j)$ is $(j + 1) \cdot TS_{2}$. If at least one preceding UN moves out the cluster, $h$ is 1 and $\Delta d_{\text{HTDMA}}(j)$ is reduced to $j \cdot TS_{2}$. There are 2j – 1 of preceding UNs' mobility status combinations in total. Based on Table 2, $\Delta d_{\text{HTDMA}}(j)$ can be expressed as

$$\Delta d_{\text{HTDMA}}(j) = TS_{2} \times \left\{ P[h = 0] \cdot (j + 1) + P[h = 1] \cdot j + \ldots + P[h = j - 2] \cdot 3 + P[h = j - 1] \cdot 2 \right\}$$

where $P[h = 0]$ is the probability that no preceding UNs move out. Other probabilities in Eq. (4) can be similarly defined. Assuming that all statuses in Table 2 are equiprobable as $\frac{1}{2j - 1}$, $\Delta d_{\text{HTDMA}}(j)$ can be simplified as $\frac{(j+2) \cdot TS_{2}}{2j - 1}$.

Accordingly, the average delay of a UN in a cluster $i$ at the $k_{a}$ TDMA2 frame in case 1, $D_{\text{HTDMA}}(k)$, can be represented as

$$D_{\text{HTDMA}}(k) = P_{\text{HTDMA}}(k) \cdot D_{\text{HTDMA}}(k) + \sum_{a=\frac{k}{2}}^{\infty} P_{\text{HTDMA}}(k) \cdot D_{\text{HTDMA}}(k).$$

First, $D_{\text{HTDMA}}(k)$ is derived as follows. As shown in Fig. 6, the UN needs to wait until the $k_{a}$ TDMA2 frame is over, which is a fixed duration because $n_{a}$ has already been determined in the $(k - 1)_{a}$ TDMA2 frame. We define this fixed waiting time duration as $d_{(j,k)}$. If all of the UNs still stay in a cluster $i$ in the $(k + 1)_{a}$ TDMA2 frame, a UN pre-allocated at the $j_{a}$ time slot cannot be allocated to an earlier time slot in the next TDMA2 frame. We consider a UN currently using the $j_{a}$ time slot. Then, the UNs for which the time slot indices are less than $j$ are defined as preceding UNs. If several preceding UNs move to another cluster, the UN using the $j_{a}$ time slot has a chance to shorten the waiting time in the next TDMA2 frame. On the other hand, although a new UN enters a cluster $i$, its movement cannot affect the time slot allocation of the UN. Thus, the waiting time in the next TDMA2 frame depends on the moving status of the preceding UNs, which is represented as $\Delta d_{\text{HTDMA}}(j)$. Finally, a UN in a cluster $i$ pre-allocated at the $j_{a}$ time slot in the $k_{a}$ TDMA2 frame needs as much as $d_{(j,k)}$ + $\Delta d_{\text{HTDMA}}(j)$ of time to send data packets at the $(k + 1)_{a}$ TDMA2 frame.

As shown in Fig. 6, $d_{(j,k)}$ is determined as $(n_{i} - j + 1) \cdot TS_{2}$. $\Delta d_{\text{HTDMA}}(j)$ depends on the mobility status of $(j - 1)$ preceding the UNs. We denote $h$ as the number of reducible TDMA2 time slots. According to the mobility status of the preceding UNs, $h$ varies, as illustrated in Table 2. For example, if all of the preceding UNs stay in a cluster $j$, $h$ is zero and $\Delta d_{\text{HTDMA}}(j)$ is $(j + 1) \cdot TS_{2}$. If at least one preceding UN moves out the cluster, $h$ is 1 and $\Delta d_{\text{HTDMA}}(j)$ is reduced to $j \cdot TS_{2}$. There are 2j – 1 of preceding UNs' mobility status combinations in total. Based on Table 2, $\Delta d_{\text{HTDMA}}(j)$ can be expressed as

$$\Delta d_{\text{HTDMA}}(j) = TS_{2} \times \left\{ P[h = 0] \cdot (j + 1) + P[h = 1] \cdot j + \ldots + P[h = j - 2] \cdot 3 + P[h = j - 1] \cdot 2 \right\}$$

where $P[h = 0]$ is the probability that no preceding UNs move out. Other probabilities in Eq. (4) can be similarly defined. Assuming that all statuses in Table 2 are equiprobable as $\frac{1}{2j - 1}$, $\Delta d_{\text{HTDMA}}(j)$ can be simplified as $\frac{(j+2) \cdot TS_{2}}{2j - 1}$.

Accordingly, the average delay of a UN in a cluster $i$ at the $k_{a}$ TDMA2 frame in case 1, $D_{\text{HTDMA}}(k)$, can be represented as

$$D_{\text{HTDMA}}(k) = P[j = 1] \cdot d_{(1,k)} + \Delta d_{(1,k)} + \ldots + P[j = 2] \cdot d_{(2,k)} + \Delta d_{(2,k)} + \ldots + P[j = n_{a} - 1] \cdot d_{(n_{a} - 1,k)} + \Delta d_{(n_{a} - 1,k)} + P[j = n_{a}] \cdot d_{(n_{a},k)} + \Delta d_{(n_{a},k)}$$

where $P[j = 1]$ is the probability that a UN is pre-allocated at the first time slot at the $k_{a}$ TDMA2 frame in a cluster $i$. Other probabilities in Eq. (5) can be also
similarly defined. If we assume that \( P[j = 1], P[j = 2], \ldots, P[j = n_k] \) are also equi-probable as \( \frac{1}{n_k} \), and \( D_{HIC}(k) \) is simplified as

\[
D_{HIC}(k) = \frac{TS}{n_k} \sum_{j=1}^{n_k} [n_k - j + 1 + \frac{(j + 3)}{2}], \quad k \geq 1. \tag{6}
\]

Second, \( D_{HIC2}(k) \) is derived as follows. As shown in Fig. 6, \( D_{HIC2}(k) \) is obtained as

\[
D_{HIC2}(k) = d_i(j, k) + \Delta d_{ad}(k+1) + \Delta d_{scp}(k+2) + \Delta d_{ad}(k+3)
\tag{7}
\]

where \( \Delta d_{ad}(k+1) \) is the delay waiting for \( BEACON \) packets during the \( (k+1)_{th} \) frame in a cluster \( ii \), \( \Delta d_{scp}(k+2) \) is the delay waiting for a \( CP \) time slot and sending its arrival to a surface station \( ii \) during the \( (k+2)_{th} \) frame, and \( \Delta d_{ad}(k+3) \) is the delay waiting for the last time slot and sending data packets during the \( (k+3)_{th} \) frame, respectively. Based on the conditions of UANets (i.e., the speed of UNs and the distance between clusters specified in Section 3), we assume that a UN can move to another cluster and listen to a new \( BEACON \) packet in tens of seconds. This implies that a UN can listen to a new \( BEACON \) packet in a cluster \( ii \) after waiting for \( d_i(j, k) + \Delta d_{ad}(k+1) \).

As illustrated in Fig. 6, we can obtain \( \Delta d_{ad}(k+1), \Delta d_{scp}(k+2) \), and \( \Delta d_{ad}(k+3) \) in Eq. (7) as

\[
\Delta d_{ad}(k+1) = (n_{i(i+1)} + 2) \times TS_2 \tag{8}
\]

\[
\Delta d_{scp}(k+2) = \left( n_{i(i+2)} + n_{i(i+3)} + 5 \right) \times TS_2 \tag{9}
\]

\[
\Delta d_{ad}(k+3) = \left( n_{i(i+3)} + 1 \right) \times TS_2 \tag{10}
\]

By substituting Eqs. (8)–(10) into Eq. (7), \( D_{HIC2}(k) \) can be expressed as

\[
D_{HIC2}(k) = TS \times [n_{i(i+1)} + n_{i(i+2)} + n_{i(i+3)} + 5], \quad k \geq 1. \tag{11}
\]

By substituting Eqs. (6) and (11) into Eq. (3) and assuming that \( P_{L_{ii}}(k) \) is equi-probable for all \( ii \) as \( 1-P_{L_{ii}}(k) \) at simplicity, \( D_{L}(k) \) can be derived as

\[
D_{L}(k) = TS \times \left\{ \frac{P_L(k)}{M} \sum_{j=1}^{n_k} \left[ n_k - j + 1 + \frac{(j + 3)}{2} \right] \right\} + \frac{1-P_{L_{ii}}(k)}{M-1} \sum_{i=1}^{M} \left[ n_{i(i+1)} + n_{i(i+2)} + n_{i(i+3)} + 5 \right]. \tag{12}
\]

### Table 2. \( h \) values according to the mobility status of preceding underwater nodes (UNs)

<table>
<thead>
<tr>
<th>Preceding UNs</th>
<th>1</th>
<th>2</th>
<th>...</th>
<th>j-2</th>
<th>j-1</th>
<th>( h )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>...</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td></td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>0</td>
<td>0</td>
<td>j-1</td>
</tr>
</tbody>
</table>

Status: 1, stay; 0, leave.
Finally, the average delay of all surface stations at the \( k \)th TDMA2 frame can be obtained by averaging Eq. (12) according to \( i \) as

\[
D_{jh}(k) = \frac{1}{M} \sum_{i=1}^{M} D_{ih}(k), \quad k \geq 1. \quad (13)
\]

2) The delay of TDMA

In case of TDMA, each surface station just pre-allocates a fixed number of time slots regardless of the mobility of UNs. Thus, a TDMA frame consists of \( N \) time slots for UNs as well as one time slot for a BEACON packet, which is used to control the UNs, including deployment of the UNs, synchronization, and termination of the application. In the worst case, the delay of TDMA is simply given as \( (N+1) \times TS_2 \). However, the average delay of TDMA, \( DT \), is obtained as

\[
\frac{(N+1) \times TS_2}{2}.
\]

3) The delay of Slotted-ALOHA

Slotted-ALOHA is similar to HTDMA in that the data packet transmission of UNs is executed during one time slot. We consider that the length of a time slot in Slotted-ALOHA is the same as \( TS_2 \), as shown in Fig. 7. However, Slotted-ALOHA is different from HTDMA in that a UN in a cluster \( i \) can only send data packets when other UNs do not attempt data packet transmission due to its random access characteristic. If a UN experiences a collision in one time slot, it tries to send data packets in the next time slot. We need to consider that the number of competing UNs in a cluster \( i \) varies according to the mobility of UNs. Thus, the delay of a UN in a cluster \( i \), \( D_{Si} \), can be represented as

\[
D_{Si} = P[s=1] \times TS_2 + [1 - P[s=1]] \times [1 - P[s=2]] \times 2 \times TS_2 + [1 - P[s=1]] \times [1 - P[s=2]] \times P[s=3] \times 3 \times TS_2 + \ldots + [1 - P[s=1]] \times [1 - P[s=2]] \times \ldots \times [1 - P[s=t-1]] \times P[s=t] \times t \times TS_2 + \ldots
\]

(14)

where \( P[s=t] \) is the probability that a UN can send data packets without collision with other UNs at the \( t \)th time slot. \( P[s=t] \) depends on the normalized traffic load flown into a UN at the \( t \)th time slot. We assume that the normalized traffic load of UNs in a cluster \( i \) at the \( t \)th time slot is the same as \( \sigma_i(s = t) \). By employing a blocking probability approach as specified in [29, 30], \( P[s=t] \) is expressed as

\[
P[s=t] = [1 - \sigma_i(s = t)]^{t-1} \quad (15)
\]

and Eq. (8) implies that \( n_i(s = t) - 1 \) UNs do not have data packets to send to a surface station \( i \) at the \( t \)th time slot. By substituting Eq. (8) into Eq. (7), we can obtain \( D_{Si} \) as

\[
D_{Si} = TS_2 \times [1 - \sigma_i(1)]^{t-1} + TS_2 \times \sum_{r=2}^{t} \left\{ S \times [1 - \sigma_i(s)]^{t-1} \times \prod_{i=2}^{r} [1 - (1 - \sigma_i(z))^{i-1}] \right\}.
\]

(16)

Similarly, the average delay of all of the surface stations employing Slotted-ALOHA, \( D_{SAi} \), is given as

\[
\frac{1}{M} \sum_{i=1}^{M} D_{Si}.
\]

C. Numerical Results

We analytically investigated the delay of HTDMA by using derived delay equations under the following conditions.

1)The diameter of a cluster governed by a surface station is given as 1,500 m with an acoustic velocity of 1,500 m/s as exemplified in [13]. Thus, the maximum propagation delay is determined to be 1 second. In addition, [13] considers the data rate of 300 bps with a packet length of 32 bytes such that the transmission delay is obtained as 0.85 second. As defined, \( TS_2 \) consists of a propagation delay, transmission delay, and some guard time for time synchronization. Finally, we fix \( TS_2 \) to 2 seconds with a guard time of 0.15 second.
We model the mobility of UNs by defining the degree of mobility, \( mp \) \((1 \leq mp \leq 10)\). It is modeled such that the number of egress and ingress UNs per time slot \( s \) depends on the value of \( mp \), which determines the probability that a UN stays in a cluster \( i \) during the frame \( k \). \( P_i(k) = 1 - 0.1 \times mp \). For example, \( mp = 1 \) implies a UN may stay in a cluster with a probability of 0.9. Namely, the higher \( mp \) is, the greater possibility that a UN moves to another cluster.

3) We consider \( \sigma_i(s) \) to be the normalized traffic load flown into a UN at a surface station \( i \) at TDMA2 time slot \( s \) assuming that the traffic load to all UNs in a cluster is equivalent at simplicity. \( \sigma_i(s) \) is considered in the range of 0.1 to 0.9 for numerical analysis.

4) \( M \) varies from 2 to 10 and \( N \) varies from 10 to 100 in order to investigate the effect of \( M \) and \( N \) on the delay performance.

Under these conditions, we compare the delay of HTDMA with that of TDMA. Fig. 8 illustrates the delays of HTDMA and TDMA according to \( M \) and \( N \) under different mobility conditions. It is shown that the delay of HTDMA decreases as \( mp \) increases, while that of TDMA is constant regardless of \( mp \). This originates from the fact that a UN moving to another cluster should try to join a new cluster by waiting for a new BEACON packet, a CP time slot, and its time slot for data transfer in the series (i.e., Eqs. (8)–(10)). The three waiting times to obtain a time slot in a new cluster results in an increase in the delay. As the increase in \( mp \) implies the increase in the number of UNs moving to other clusters, a UN can experience a rise in the delay. On the other hand, the delay of HTDMA decreases as \( M \) increases. This implies that HTDMA can be more efficient with respect to the delay by sharing mobility information among surface stations. As a result, HTDMA can outperform TDMA case by case. In the case that the mobility is weak \((mp \leq 5)\) and \( M \geq 4 \), the delay of HTDMA is shorter than that of TDMA, regardless of \( N \). In the case that the mobility is strong \((mp > 5)\), the delay of HTDMA is shorter than that of TDMA when \( M > 4 \) regardless of \( N \). Thus, it is shown that HTDMA can be more efficient than TDMA when the number of clusters increases.

We also compare the delay of HTDMA with that of Slotted-ALOHA. As illustrated in Fig. 9, the delay of Slotted-ALOHA also decreases as \( M \) increases just as with HTDMA. This is because the number of UNs per cluster decreases such that contention among UNs in a cluster becomes weak when \( M \) increases; the delay of Slotted-ALOHA also decreases. However, as the traffic loads
increase, Slotted-ALOHA results in unbounded delay, while HTDMA supports a consistent delay regardless of traffic loads by pre-assigning a time slot for a specific UN. In addition, the delay of Slotted-ALOHA increases as \( N \) increases because of severe contention among UNs in a cluster. As a result, the delay of HTDMA is shorter than that of Slotted-ALOHA in case of \( M \leq 4 \) and \( \sigma(s) \geq 0.5 \). Otherwise, HTDMA can also outperform Slotted-ALOHA in the case that Slotted-ALOHA experiences severe contention such as an increase in the traffic load and \( N \), as well as a decrease in \( M \).

VI. DISCUSSION AND CONCLUSIONS

In this paper, we propose a new MAC protocol, referred to as HTDMA, which is composed of TDMA1 and TDMA2. A major function of TDMA1 is to cover diverse on-demand UANet applications and forward data from these applications to a terrestrial base station. TDMA2 is separately executed in all clusters once a specific application is requested. The number of time slots in each cluster is managed by all of the surface stations, which shares mobility information using TDMA1. Then, each surface station allocates time slots into UNs currently residing in its cluster using TDMA2. With the help of HTDMA, we can accommodate multiple UANet applications simultaneously, and also improve delay performance by efficiently controlling the mobility of UNs. It has been analytically shown that HTDMA can reduce the delay of TDMA, particularly in the case that the number of clusters increases. In addition, HTDMA can outperform Slotted-ALOHA in terms of delay in the case of severe contention environments such as heavily loaded traffic under an increase in the number of UNs in a cluster. This implies that HTDMA can reduce the possibility of dropped data in the queue or lost data in the channel by decreasing the overall network delay. Furthermore, HTDMA is expected to improve channel efficiency by avoiding the occurrence of idle time slots induced by UNs’ mobility. Numerical results can be displayed in an engineering table, which should be helpful for designing a UANet. Finally, delay analysis considering mobility modeling could be employed in other MAC protocols for a UANet such as pure ALOHA and extended to other network scenarios.
ACKNOWLEDGMENTS

This work was conducted as a part of the research projects of “Development of the wide-area underwater mobile communication systems” financially supported by the Ministry of Land, Transport and Maritime Affairs of Korea.

REFERENCES


Changho Yun
was born in Busan, South Korea in 1976. He received the M.S. and Ph.D. degrees in the School of Information and Communications from Gwangju Institute of Science and Technology (GIST) in 2004 and 2007, respectively. He was then a post-doctoral researcher in the Department of Computer Science in North Carolina State University. He is a senior researcher in the Department of Ocean Engineering Research of the Korea Institute of Ocean Science and Technology (KIOST). His current research interests include underwater acoustic networks (UANets) and maritime communication networks.

A-Ra Cho
was born in Jinhae, South Korea in 1979. She received the Ph.D. degree the Department of Underwater Acoustic Communication Engineering at the University of Science and Technology (UST) in 2012. She is currently a post-doc researcher in the Department of Ocean Engineering Research of the Korea Institute of Ocean Science and Technology (KIOST). Her current research interests include mobile ad-hoc networks (MANETs) and underwater acoustic networks (UANets).

Seung-Geun Kim
received the B.S. degree in Electronic Engineering from Inha University, Incheon, Korea, in 1995, and the M.S. and Ph.D. degrees in Information and Communications from Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, in 1997 and 2002, respectively. Since 2002, he has worked for the Korea Institute of Ocean Science and Technology (KIOST) as a principal researcher. His research interests include acoustic underwater communication systems and maritime communication system design and implementation.

Jong-Won Park
was born in Hwasung, South Korea in 1973. He received the M.S. and Ph. D. degrees from the Department of Electronic Engineering of Ajou University in 1997 and 2006, respectively. Currently, he is a principal researcher in the Department of Ocean Engineering Research of the Korea Institute of Ocean Science and Technology (KIOST). His current research interests include underwater robot (AUV, ROV) systems, maritime logistics, ship navigation systems, integrated networks for ships and naval vessels, as well as underwater acoustic communications and networks (UANets).

Yong-Kon Lim
received an M.S. degree in power and electrical engineering from Chung-Nam University in 1984 and a Ph.D. degree in electronic engineering from A-Jou University in 1994. He has been working at the Korea Institute of Ocean Science and Technology (KIOST) since 1980. Presently, he is a principal researcher and a professor in the Department of Underwater Acoustic Communication Engineering at the University of Science and Technology (UST) and an affiliate professor in that of Ocean Systems Engineering (OSE) at the Korea Advanced Institute of Science and Technology (KAIST). His main interests are underwater acoustic telecommunication, underwater acoustic networks, integrated logistics systems for the ocean, and integrated communication systems for ships.