Journal of Engineering and Applied Sciences 13 (21): 9132-9140, 2018
ISSN: 1816-949X

TDMA Based Collision Free MAC Protocol in Underwater Acoustic Networks

Muzaffar Ergashev and Sunmyeng Kim
Department of Software Engineering, Kumoh National Institute of Technology,
730-701 Gumi, South Korea

Abstract: Number of MAC protocols have been proposed for Underwater Acoustic Networks (UANs). However, unique characteristics of acoustic communication such as very slow data rate inconsistent long propagation delay and long physical header degrade the performance of the most existing works. In order to improve network performance, COD-TS has been proposed. COD-TS utilizes clustering algorithm to organize nodes into clusters. To avoid inter-cluster collisions, cluster heads exchange schedule information between them based on CDMA. COD-TS has two problems. Near-far effect has not been considered for schedule information packets. It exchanges schedule information at high power, causing collisions to member nodes in the neighboring cluster. Therefore, it causes degradation of network performance. We introduce a cluster based underwater TDMA protocol (CU-TDMA) which is collision free and energy efficient. In the CU-TDMA, first an UAN is partitioned into layers based on water pressure. Then, nodes in each layer are organized into clusters. To address the inter-layer inter-cluster and intra-cluster collisions, we present a TDMA algorithm. We compare the performance of the proposed protocol with the COD-TS in simulations. The results show that the proposed protocol outperforms the COD-TS protocol in terms of throughput and energy efficiency.

Key words: Underwater acoustic networks, clustering, collision free, layering, time slot, COD-TS

INTRODUCTION

Underwater Acoustic Networks (UANs) are very effective system for exploring and observing under oceans. UANs are the system which consists of sensor nodes, Autonomous Underwater Vehicles (AUV) and floating buoys. They utilize acoustic waveforms for communication and do special tasks. Main purposes of UANs are: ocean sampling, environmental monitoring, undersea explorations, disaster prevention assisted navigation, distributed tactical surveillance and mine reconnaissance (Chen *et al*., 2014; Akyildiz *et al*., 2005).

Terrestrial networks mainly use two types of waveforms: electromagnetic waves and optical waves. Electromagnetic waves are easily absorbed in salty water. Thus, they propagate only for short distance in water and additionally require large size antennas. Optical waves do not suffer much attenuation in water but are affected by scattering. Also, optical waves require sender and receiver's devices to be well directed to each other. Because of above mentioned reasons, both electromagnetic and optical waves are not suitable for vunderwater communication. Differently from electromagnetic and optical waves, acoustic waves propagate long distance and are not affected by scattering. Therefore, acoustic waves are utilized for underwater communications. Acoustic waveforms have unique features such as low propagation speed, multipath effect, high bit error rate and narrow frequency bandwidth. The features make great challenges for researchers of UANs. To address the challenges and improve network performance, numerous MAC protocols for UANs have been proposed. Currently, there are three types of MAC protocols: contention, contention free and hybrid MAC protocols (Chen *et al*., 2014).

Contention MAC protocols are based on random access to the channel. Slotted-FAMA divides the channel into time slots (Molins and Stojanovic, 2007). Packets are sent only at the beginning of each slot. To improve channel utilization, UW-Flashr allows sending multiple packets after reserving the channel (Yackoski and Shen, 2008). UPMAC protocol (Zhu *et al*., 2016) uses ALOHA protocol in low traffic and it switches to the three-way handshake in high traffic load. Due to the long preamble and low transmission speed of acoustic communication degrades efficiency of contention or handshaking MAC protocols (Zhu *et al*., 2015).

Corresponding Author: Sunmyeng Kim, Department of Software Engineering, Kumoh National Institute of Technology, 730-701 Gumi, South Korea

Contention free MAC protocols divides channel resource into partitions by frequency (FDMA), time (TDMA) or pseudo-random codes (CDMA) and then allocate the partitions to each node in the network. A transmitter-based OFDMA scheme (UW-OFDMAC) was proposed which includes OFDMA parameters self-assignment algorithm (Bouabdallah and Boutaba, 2011). Due to the limited bandwidth of underwater acoustic channels, the FDMA protocols cannot improve network throughput much. In POCA-CDMAMAC protocol, the nodes in the same path are assigned with the same spreading sequence and they transmit their packets in a round-robin way (Fan *et al*., 2011). In CDMA protocols, the number of codes is limited to give an identical code for each node in a large size of network. In ST-MAC, the base station collects required information and computes the transmission/reception schedule for each sensor node in the network (Hsu *et al*., 2009). STUMP utilizes node position diversity to increase channel utilization (Kredo II *et al*., 2009). Clustering based MAC protocol called COD-TS has been proposed by Zhu *et al*. (2015). The cluster heads schedule transmissions of member nodes and exchange up-to-date schedule using CDMA. In COD-TS, schedule packets are transmitted at higher transmission power and that causes collision at neighboring cluster's members. Also, the well-known near-far effect has not been taken into account. Those shortcomings of the protocol decrease overall performance of the network.

Hybrid MAC protocols utilize contention and contention free protocols to combine advantages of both. Hybrid MAC protocols have been presented by Kredo II and Mohapatra (2007), Namgung *et al*. (2010), Pompili *et al*. (2009) and Tan and Seah (2007). H-MAC protocol divides a time frame into two time slots, one is used for contention free scheme and the other one is for random access (Kredo II and Mohapatra, 2007). P-MAC protocol consists of a contention-free protocol and slotted MACA (Namgung *et al*., 2010). In UW-MAC protocol, the signaling packets are sent by the ALOHA scheme then, data packets are sent by the CDMA (Pompili *et al*., 2009). PLAN protocol (Tan and Seah, 2007) combines the CDMA and MACA schemes. In those hybrid MAC protocols, high collision rate is inherent from contention based protocols and they are not scalable for a large size of network due to limitation of (bandwidth or code) resources.

To increase energy efficiency and network throughput, we propose a cluster based underwater TDMA protocol (CU-TDMA). In CU-TDMA, first, network is partitioned into layers based on water pressure measured by sensor nodes. Then, nodes in each layer are organized into clusters. Channel time is divided into time slots. Time slots are assigned to clusters. Cluster heads schedule transmissions of their member nodes. Therefore, transmissions do not collide.

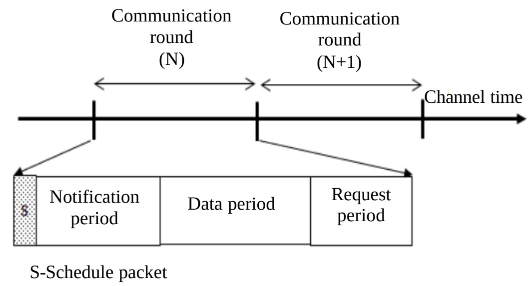

Fig. 1: Channel time in COD-TS

COD-TS protocol: In this study, we review COD-TS protocol. For synchronizing nodes, COD-TS uses algorithms presented by Liu *et al*. (2010) and Lu *et al*. (2010). Also, COD-TS organizes nodes into clusters using algorithm proposed by Bandyopadhyay and Coyle (2003). A cluster consists of a cluster head node and one hop neighbor nodes (member nodes). Channel time is divided into communication rounds (Fig. 1). Duration of a communication round dynamically changes according to network load. A communication round consists of notification period, data transmission period and request period. At the beginning of the nth communication round, a cluster head makes a new schedule for the nth notification period and data transmission period based on requests from member nodes in the (n-1)th request period. It also, makes schedules for request transmissions for the nth request period. When a cluster head makes a schedule, it considers the schedule information of neighbor clusters to avoid collisions. A cluster head begins nth communication round by transmitting a schedule packet to the neighbor cluster heads. Then, it sends notification packets to the members. In the data transmission period, member nodes send their data packets according to notification packets. In the request period when a member node has data packets to send, it transmits a request packet to the cluster head.

Because of distance between cluster heads inter cluster synchronization error is more severe compared to intra cluster synchronization error. For compensating inter cluster synchronization error, longer guard time is required. The guard time for schedule exchange packets causes large overhead. Also, COD-TS uses CDMA to transmit schedule packets. A cluster head may not exactly decode the packets sent from its neighbor cluster head because of near-far effect in CDMA. In this case, the cluster head cannot consider the schedule of the neighbor cluster and make its own schedule only based on the requests sent from its member nodes. Consequently, this may cause collisions among data packets sent from nodes of neighbor clusters. Additionally, the schedule packet itself collides with transmissions of member nodes of the neighboring clusters. Collided packets should be retransmitted. This degrades energy efficiency and prolongs end to end delay of the packets. Transmission of schedule packets at high power decreases battery life which leads network "Holes" after the battery is depleted.

To overcome shortcomings mentioned before we propose a novel TDMA based MAC protocol called CU-TDMA. CU-TDMA does not use exchanging schedule information between neighbor clusters.

CU-TDMA protocol: The proposed CU-TDMA protocol is based on cluster and TDMA to avoid packet collisions. In the proposed protocol, channel time is divided into time slots. The CU-TDMA protocol organizes sensor nodes into clusters and then allocates a time slot to each cluster. A cluster head in a cluster schedules its member nodes for transmitting data packets.

In order to allocate a time slot to each cluster, we use three variables: LAYER, HOP AND INDEX. A 3 Dimensions (3D) underwater network is divided into 2D layers according to the depth of water. The LAYER means the layer number to which each cluster belongs. The HOP is the distance from the center cluster to a cluster. The INDEX is the index number of the clusters with the same HOP. To get these three values, the CU-TDMA protocol layers sensor nodes in the network and then organizes clusters in a tree structure.

Layering of sensor nodes in the network: An underwater sensor network consists of a surface buoy, on-shore data center and sensor nodes (Fig. 2). Sensor nodes collect data and transmit them to the surface buoy. The surface buoy forwards them to the on-shore data center. The surface buoy includes an antenna for RF transmission.

All sensor nodes are randomly deployed in three Dimensions (3D). A 3D underwater network is divided into 2D layers according to the depth of water. We intuitively set the distance between layers as 2·r.r is the transmission range. We denote the maximum depth of water as DEPTH max. The number of 2D layers (N) is as following:

$$N = \left\lceil \frac{DEPTH_{max}}{2.r} \right\rceil \quad (1)$$

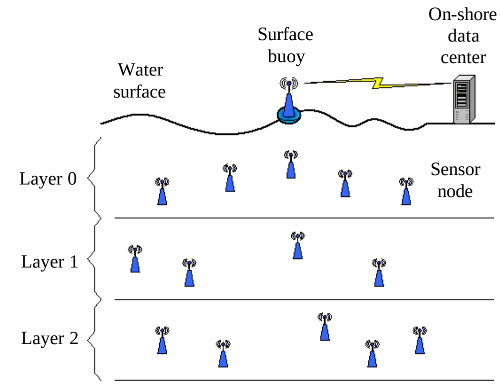

Fig. 2: Topology of UANs

where, $\lceil x \rceil$ is the ceiling function. The layer near the surface of the water is numbered 0 and the deepest layer is numbered N-1.

Each node measures water pressure (p) by using a pressure gauge. And it obtains its depth from the pressure as following (Anonymous, 2018):

$$DEPTH = \frac{\rho}{\rho.g} \quad (2)$$

where, ρ is the fluid density (kg/m^3) and g is gravitational acceleration (m/sec^2). A node i belongs to Layer k (= 0, 1, 2, ..., N-1) if its depth (DEPTHi) satisfies a following condition:

$$2.k.r < DEPTHi < 2.(k+1).r \quad (3)$$

Figure 2 shows an example of layering nodes in UANs. We assume that maximum depth DEPTHmax is 6600 m and transmission range r is 1100 m. The number of layers is (3 = 6600/(2·1100)). We consider three nodes (N_1-N_3) with depths of 800, 4100 and 5200 m, respectively. Therefore, the nodes N_1-N_3 belong to Layers 0, 1 and 2, respectively. The CU-TDMA protocol can get the LAYER value through layering of sensor nodes.

Organizing clusters in a tree structure: After layering, nodes in each layer are organized into clusters by using a clustering algorithm proposed in 2018. Clustering algorithms are beyond the scope of this researech. We assume that all nodes in each layer are well clustered.

The clusters are organized in a tree structure. A tree structure organizes clusters hierarchically. It is a collection of entities called clusters. Clusters are connected by edges. Each cluster may or may not have a child cluster.

Here, we show how to organize clusters in a tree structure. Each layer has one tree structure. The cluster at the top of the tree is called root. There is only one root cluster in each layer. The root cluster of layer N is determined by the root cluster of Layer N-1. The root cluster of layer N-1 determines the cluster with the closest distance to itself among the clusters in layer N as the root cluster. The root cluster of Layer 1 is determined by the surface buoy.

Each cluster has a parameter (HOP) which is the number of edges between itself and the root cluster. The HOP starts from 0. The root cluster of the tree structure has HOP 0 and the HOP of any other clusters in the tree is one more than the HOP of its parent.

In order to create a tree structure, we use a tree creation packet which includes mainly two parameters: the value of HOP and address of parent. The root cluster in each layer sends tree creation packets with an HOP value of zero and a parent of zero to its neighboring clusters. If the address of parent is 0, it means that there is no parent. After receiving a tree creation packet, a cluster increases the HOP value in the packet by 1 and sets it as its own HOP value. And then, the cluster makes another tree creation packet with its own HOP value and address of its parent and sends it to its neighboring clusters. This process is repeated until the packet reaches leaf clusters.

A cluster can receive multiple tree creation packets from its neighboring clusters. Whenever, a cluster receives a tree creation packet from another cluster, it decides the relation of its neighboring clusters and the value of its HOP based on Algorthim 1. The relation is one of PARENT, CHILD or SIBLING. If a cluster i receives a tree creation packet from a cluster j with the lower HOP value, it sets its HOP value to HOPj+1, chooses the cluster j as its parent (i.e., the relation is set to PARENT). If the HOP values of the clusters i and j are the same, the relation is set to SIBLING. If the cluster j chooses the cluster i as its parent, the relation is set to CHILD.

Algorthim 1; Algorithm for deciding the values of HOP and relation:

```
Cluster i Information: HOPi, Parenti, Relationj
Cluster j Information: HOPj, Parentj
If (HOPi>HOPj) {
    HOPi = HOPj+1
    Parenti = j
    Relationj = PARENT
} else if (HOPi == HOPj) {
    Relationj = SIBLING
} else if (HOPi<HOPj) {
    if (i == Parentj) {
        Relationj = CHILD
  }
}
```

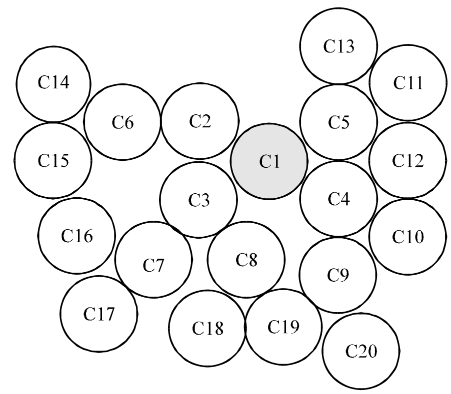

Fig. 3: Topology for creating a tree structure

If a cluster receives multiple tree creation packets from the clusters with the same value of HOP, it selects a cluster with the largest RSSI (Received Signal Strength Indicator) as its parent.

Each cluster has up to two siblings. If a cluster has more than two siblings, it selects two siblings with the largest RSSI.

After organizing a tree structure, each cluster allocates index numbers which starts from 0. To do this, we use a sibling index packet. A sibling index packet has three fields: ID, NS (number of siblings) inDEX. The ID field is the cluster ID, NS is the number of its siblings and INDEX is the sibling index. If a cluster has the smallest ID among its sibling clusters, it creates a sibling index packet and transmits the packet to its siblings. Otherwise, it just waits for sibling index packets from its siblings. When a cluster receives a sibling index packet, it sets its own index number to (INDEX+1). And then, it increases the INDEX value in the received sibling index packet by 1 and forwards the packet to its next sibling.

If a cluster receives multiple sibling index packets, it sets its index number based on following criteria:

- Criterion 1: sibling index packet with the smallest NS value
- Criterion 2: sibling index packet with the smallest ID value

We show an example of creating a tree structure and assigning an index number. In Fig. 3, there are 20 clusters and the cluster C1 is chosen as the root cluster. The cluster C1 sends a tree creation packet (HOP: 0, Parent: 0) to its neighbor clusters C2-C5. When they receive the tree creation packet, they set their HOP values to 1 and choose the cluster C1 as their parent cluster. And then they make a tree creation packet (HOP: 1, Parent: C1) to its neighbor clusters (parent and siblings). When the

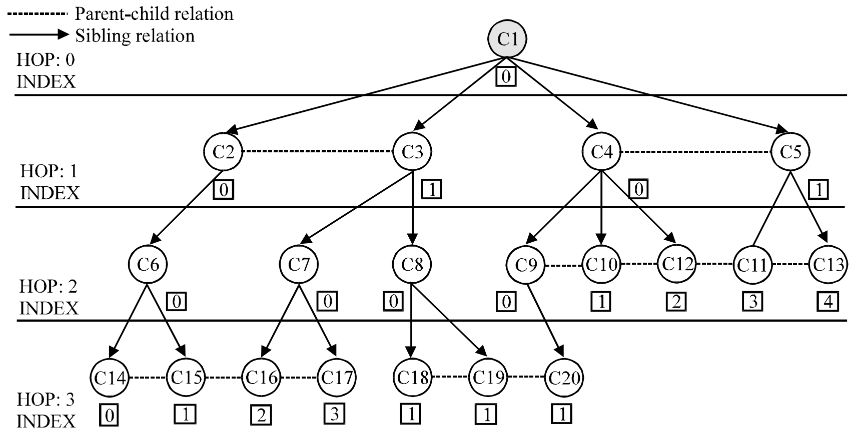

Fig. 4: Example of tree structure creation

Packet source	C9	C10	C12	C11	C13
C9	0	1	2	3	4
C11	3	2	1	0	1

Fig. 5: Index numbers

cluster C1 receives the packets, it becomes their parent. When the clusters C2 and C3 receive the packets from each other, they become sibling of each other since they have the same value of HOP, (i.e., 1). This is similar to the clusters C4 and C5. The other clusters also repeat this process.

Figure 4 shows the completed tree structure for the topology in Fig. 4. We can see that the HOP of cluster C1 is 0, the HOP of cluster C2~C5 is 1 and the HOP of cluster C6~C13 is 2 and the HOP of cluster C14~C20 is 3.

After organizing the tree structure, each cluster allocates index numbers. We show an example of assigning index numbers to clusters C9-C13. Each cluster checks whether its ID is the smallest among its siblings. Since, the IDs of the clusters C9 and C11 are the lowest, they make sibling index packets ([ID: C9, NS: 1 INDEX: 0] for the cluster C9, [ID: C11, NS: 2 INDEX: 0] for the cluster C11) and transmit the packets to their siblings. After receiving the packets, the siblings set their index numbers to (INDEX+1). This process repeats until the packets reach the last sibling clusters.

Figure 5 shows the index numbers assigned based on the sibling index packets from the clusters C9 and C11. The packet from the cluster C9 includes a smaller NS value, all the clusters choose its index number assigned based on the packet from the cluster C9. From Fig. 4, we can see the index numbers assigned to each cluster.

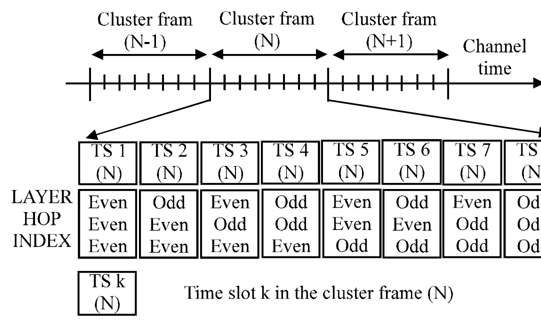

Fig. 6: Time slot assignment

Allocating a time slot to each cluster In the proposed protocol, channel time is divided into cluster frames (Fig. 6). Each cluster frame has 8 Time Slots (TSs).

Time slots allocated to the cluster are re-used in a regular pattern. To ensure that the mutual interference between clusters remains below a harmful level, adjacent clusters use different time slots. The closest distance between two cluster using the same time slot is called the time slot reuse distance. Figure7 shows, if the clusters C1 and C2 use the same time slot, then the reuse distance is one. If the clusters C1 and C3 use the same time slot, then the reuse distance is two. Wen neighboring clusters use the same time slot (reuse distance = 1), they cause interference each other. In order to avoid interference, it is better to use a distance value of 2 or more. The proposed CU-TDMA protocol uses the distance value of 2 to get the best channel utilization.

We use three variables to allocating a time slot to each cluster. Each variable value is divided into odd number and even number, since, the time slot reuse distance is two. Therefore, the CU-TDMA protocol needs

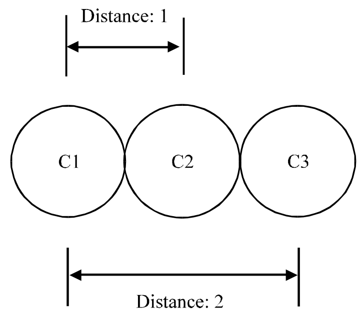

Fig. 7: Time slot reuse distance

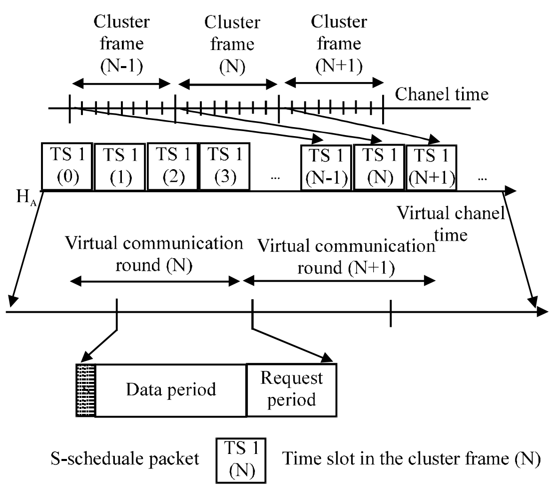

Fig. 8: Virtual channel time

($8 = 2^3$) time slots per each cluster frame. Each cluster determines its own time slot based on its LAYER, HOP and INDEX values (Fig. 6).

Scheduling member nodes in a cluster: A cluster head assembles its time slots from cluster frames to the virtual channel time as illustrated in Fig. 8. Then, the cluster head separates virtual channel time into the Virtual Communication Rounds (VCRs). Each VCR consists of a data period and a request period. The cluster head begins a new VCR by sending a schedule packet.

At the beginning of the nth virtual communication round, a cluster head makes a new schedule for the nth data period and request period. The schedule for the data period is based on requests from member nodes in the (n-1)th request period. The schedule for the request period is for transmitting request packets. The schedule includes information about which member node transmits data or request packets at which time slot. After making a schedule packet, the cluster head transmits it to its member nodes. After receiving the schedule packet, each member node starts the transmission period and transmits its data packets at the time slot assigned to itself. In CU-TDMA protocol, a member node piggybacks its request packets to its data packets.

In the request period, if a member node has data packets to send, then it transmits a request packet at the time slot assigned to itself. Otherwise, it does not do anything. A request packet includes information on the number of data time slots that the member node needs in the next VCR.

Figure 9 shows an example of packet transmissions in the proposed CU-TDMA protocol. There are a cluster Head (H) and two member nodes (N_1 and N_2). In the request period of VCR N-1, the cluster head H receives request packets from its member nodes N_1 and N_2. We assume that the member nodes N_1 and N\ requests two data time slots and one data time slot, respectively. At the beginning of VCR N, the cluster head H allocates three data time slots for data packets and then sends the schedule packet to its member nodes. The schedule packet does not include time slots for request packets, since, the member nodes may piggyback their request packets at the data time slots. The member node N_2 transmits its data packets with a request packet (one data time slot request). However, the member node N_1 only transmits its data packet because it has no data to transmit anymore. At the beginning of VCR N+1, the cluster head H sends a schedule packet including one data time slot for the member node N_2 and one request time slot for the member node N_1.

Performance evaluation: In this study, we compare COD-TS and CU-TDMA protocols through simulation using Network Simulator NS-3. Simulation is done for 3D topology of multi-hop UANs.

For both COD-TS and CU-TDMA, 160 sensor nodes were randomly distributed in the 3D network of 6600×5800×5400 m. Each cluster consists of 8 sensor nodes including a cluster head. Therefore, there 20 clusters in the network. We assume that transmission range is 1100 m and transmission rate is 667 bps. Packet preamble is added with duration of 1.5sec by the physical layer. Propagation speed of acoustic carrier is set to 1500 m/sec. Transmission power is set to 2 W and power consumptions of a node in reception and idle states are both set to 158 mW. For COD-TS, the transmission power for the schedule packet is set to 5 W. The length of a data packet is 400 bytes. We use the negative exponential distribution to get the lengths of the data packet

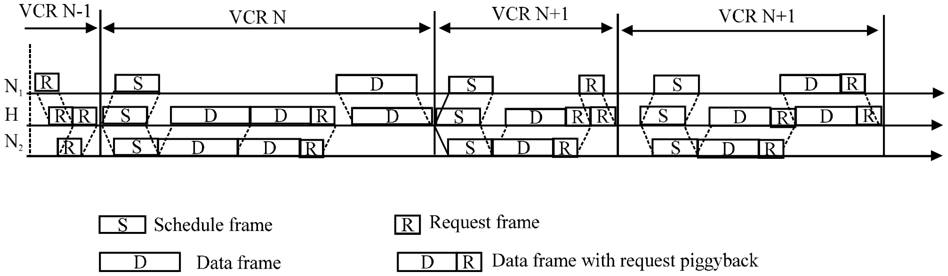

Fig. 9: An example of virtual communication rounds

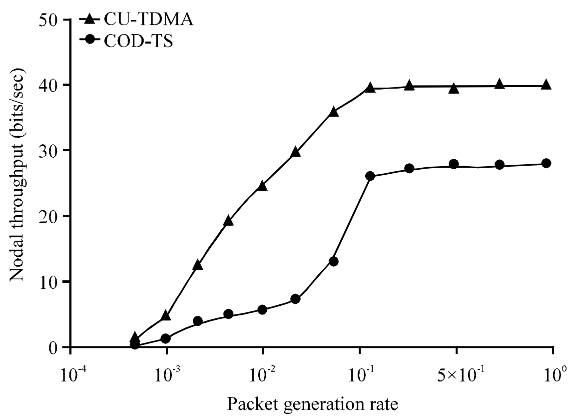

Fig. 10: Nodal throughput according to the data packet generation rate

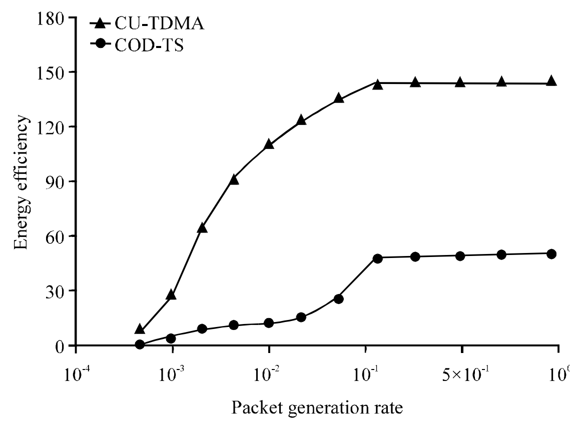

Fig. 11: Energy efficiency according to the data packet generation rate

inter-arrival times. The average inter-arrival time (packet generation rate) of the distribution with arrival rate parameter λ is $1/\lambda$. Each node is allowed to send at most 4 frames in a round. Queue size of each node is set as much as big enough to prevent overflow of frames. Simulation run time is 10000 sec. Simulation results are averaged over 20 simulation runs.

The main performance metrics of interest are nodal throughput and energy efficiency. Nodal throughput is an average number of successfully received bits in a second per node. Energy efficiency is an average number of successfully received bits per joule of total used energy.

Figure 10 and 11 show the effect of the packet generation rate on the nodal throughput and energy efficiency. As shown in Fig. 10, the proposed CU-TDMA protocol achieves higher nodal throughput than the COD-TS protocol. The COD-TS protocol gets very low nodal throughput at low packet generation rate. This is because the COD-TS protocol spends more time for sending schedule packets frequently compared to data packet transmissions. Also, the schedule packet collides with transmissions of nodes in neighboring clusters. On the contrary, the nodal throughput of the CU-TDMA protocol linearly increases at low packet generation rates and it switches to saturation state at high packet generation rates.

Figure 11 demonstrates the energy efficiency performance. The CU-TDMA protocol achieves almost three times more energy efficiency than the COD-TS protocol. The reason for this is similar to the reason explained for nodal throughput. The COD-TS protocol transmits schedule packets at higher power level. As mentioned above schedule packets cause collisions to nodes in neighboring clusters. And data packets may collide because of incorrectly received schedule packets. Therefore, energy efficiency decreases.

Figure 12 and 13 show the effect of the data packet size on network performance. We can see that the proposed CU-DMA protocol always has better performance than the COD-TS protocol. Throughput and energy efficiency of both protocols are getting better when the data packet size increases. This is because the proportion of control packet (overhead) size decreases as the data packet size increases.

Figure 14 and 15 show the effect of the number of sensor nodes on the nodal throughput and energy efficiency. When the number of sensor nodes increases, the number of member nodes included in one cluster

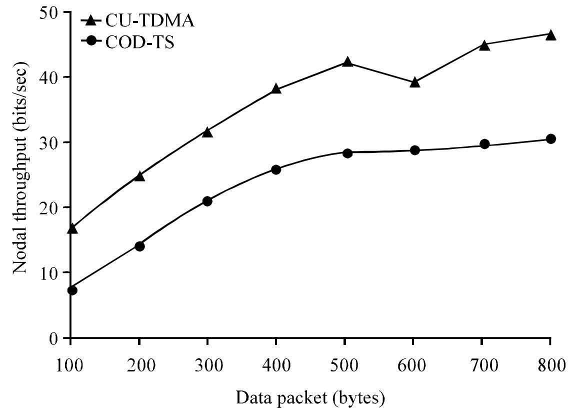

Fig. 12: Nodal throughput according to the data packet size

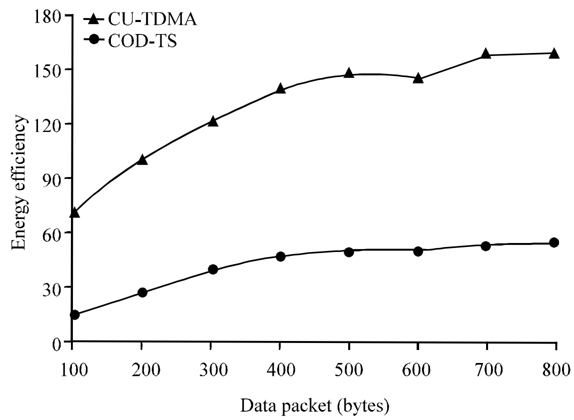

Fig. 13: Energy efficiency according to the data packet size

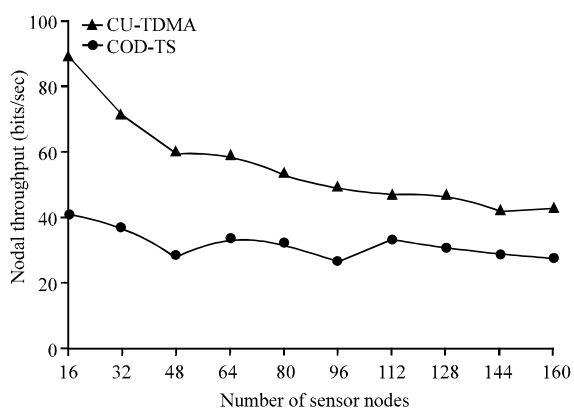

Fig. 14: Nodal throughput according to the number of sensor nodes

increases. Therefore, the channel time that a cluster head allocates to a member node is reduced. Consequently, the network performance are getting lower.

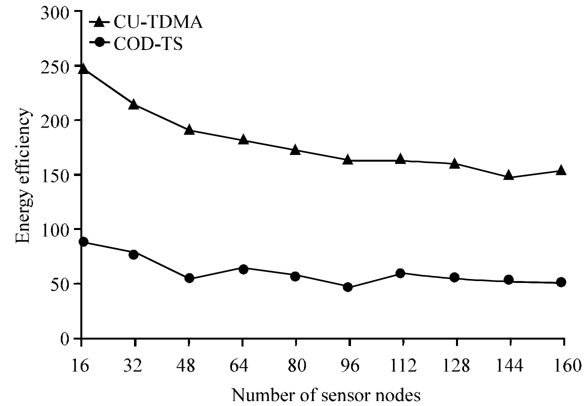

Fig. 15: Energy efficiency according to the number of sensor nodes

CONCLUSION

Uunderwater Acoustic Networks (UANs) have low network performance, since, unique characteristics of acoustic communication such as very slow data rate inconsistent long propagation delay and long physical header. In order to improve network performance, the COD-TS protocol was proposed. The COD-TS protocol organizes sensor nodes into clusters. To avoid inter-cluster collisions, cluster heads exchange schedule information. The COD-TS protocol still has low performance. We proposed a new MAC protocol, called CU-TDMA which is collision free and energy efficient. In the CU-TDMA, a network is partitioned into layers based on water pressure. Then, nodes in each layer are organized into clusters. To address the inter-layer inter-cluster and intra-cluster collisions, we present a time slot allocation algorithm by using a tree structure. Performance evaluation is conducted using simulation and shows that the proposed protocol significantly outperforms the previous protocol in terms of nodal throughput and energy efficiency.

ACKNOWLEDGEMENT

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

REFERENCES

Akyildiz, I.F., D. Pompili and T. Melodia, 2005. Underwater acoustic sensor networks: Research challenges. Ad Hoc Networks, 3: 257-279.

Anonymous, 2018. Hydrostatics. Wikipedia, San Francisco, California, USA. https://en. wikipedia. org/wiki/Hydrostatics#Pressure_in_fluids_at_rest

Bandyopadhyay, S. and E.J. Coyle, 2003. An energy efficient hierarchical clustering algorithm for wireless sensor networks. Proceedings of the 22nd Annual Joint Conference of the Computer and Communications, Volume 3, March 30-April 3, 2003, San Franciso, CA., USA., pp: 1713-1723.

Bouabdallah, F. and R. Boutaba, 2011. A distributed OFDMA medium access control for underwater acoustic sensors networks. Proceedings of the 2011 IEEE International Conference on Communications (ICC'11), June 5-9, 2011, IEEE, Kyoto, Japan, ISBN:978-1-61284-232-5, pp: 1-5.

Chen, K., M. Ma, E. Cheng, F. Yuan and W. Su, 2014. A survey on MAC protocols for underwater wireless sensor networks. IEEE. Commun. Surv. Tutorials, 16: 1433-1447.

Fan, G., H. Chen, L. Xie and K. Wang, 2011. An improved CDMA-based MAC protocol for underwater acoustic wireless sensor networks. Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM'11), September 23-25, 2011, IEEE, Wuhan, China, ISBN:978-1-4244-6250-6, pp: 1-4.

Hsu, C.C., K.F. Lai, C.F. Chou and K.J. Lin, 2009. ST-MAC: Spatial-temporal mac scheduling for underwater sensor networks. Proceedings of the 2009 IEEE Conference on INFOCOM, April 19-25, 2009, IEEE, Rio de Janeiro, Brazil, ISBN: 978-1-4244-3512-8, pp: 1827-1835.

Kredo II, K., P. Djukic and P. Mohapatra, 2009. STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol. Proceedings of the 2009 IEEE Conference on INFOCOM, April 19-25, 2009, IEEE, Rio de Janeiro, Brazil, ISBN: 978-1-4244-3512-8, pp: 2961-2965.

Kredo II, K.B. and P. Mohapatra, 2007. A hybrid medium access control protocol for underwater wireless networks. Proceedings of the 2nd Workshop on Underwater Networks, September 14, 2007, ACM, Montreal, Quebec, Canada, ISBN:978-1-59593-736-0, pp: 33-40.

Liu, J., Z. Zhou, Z. Peng and J.H. Cui, 2010. Mobi-Sync: Efficient time synchronization for mobile underwater sensor networks. Proceedings of the 2010 IEEE Conference on Global Telecommunications (GLOBECOM'10), December 6-10, 2010, IEEE, Miami, Florida, ISBN:978-1-4244-5636-9, pp: 1-5.

Lu, F., D. Mirza and C. Schurgers, 2010. D-sync: Doppler-based time synchronization for mobile underwater sensor networks. Proceedings of the 5th ACM International Workshop on Under Water Networks, September 30-October 01, 2010, ACM, Woods Hole, Massachusetts, ISBN:978-1-4503-0402-3, pp: 1-8.

Molins, M. and M. Stojanovic, 2007. Slotted FAMA: A MAC protocol for underwater acoustic networks. Proceedings of the 2006-Asia Pacific Conference on OCEANS, May 16-19, 2007, IEEE, Singapore, ISBN:978-1-4244-0137-6, pp: 1-7.

Namgung, J.I., S.Y. Shin, N.Y. Yun and S.H. Park, 2010. Adaptive GTS allocation scheme based on IEEE 802.15.4 for underwater acoustic sensor networks. Proceedings of the 2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC'10), December 11-13, 2010, IEEE, Hong Kong, China, ISBN:978-1-4244-9719-5, pp: 297-301.

Pompili, D., T. Melodia and I.F. Akyildiz, 2009. A CDMA-based medium access control for underwater acoustic sensor networks. IEEE. Trans. Wirel. Commun., 8: 1899-1909.

Tan, H.X. and W.K. Seah, 2007. Distributed CDMA-based MAC protocol for underwater sensor networks. Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN'07), October 15-18, 2007, IEEE, Dublin, Ireland, pp: 26-36.

Yackoski, J. and C.C. Shen, 2008. UW-FLASHR: Achieving high channel utilization in a time-based acoustic MAC protocol. Proceedings of the Third ACM International Workshop on Underwater Networks, September 14-19, 2008, ACM, San Francisco, California, USA., ISBN: 978-1-60558-185-9, pp: 59-66.

Zhu, M., W. Zhang, N. Jin, Z. Qin and J. Xin *et al*., 2016. UPMAC: A localized load-adaptive MAC protocol for underwater acoustic networks. IEEE. Sens. J., 16: 4110-4118.

Zhu, Y., Z. Peng, J.H. Cui and H. Chen, 2015. Toward practical MAC design for underwater acoustic networks. IEEE. Trans. Mobile Comput., 14: 872-886.