Cross Layer Based Cooperative Communication Protocol for Improving Network Performance in Underwater Sensor Networks

Kihyun Kim, Sunmyeng Kim[*]

Department of Computer Software Engineering, Kumoh National Institute of Technology, Korea

Abstract

For underwater sensor networks (USNs), cooperative communications have been introduced to improve network performance with the help of relay nodes. The previous cooperative communications select the best relay node on a hop-by-hop basis. Therefore, they have limitations in improving performance. In order to get better performance, a cooperative communication protocol based on the cross layer is proposed in this paper. The proposed protocol uses the information provided by a routing protocol at the network layer for the erroneous data packet delivery. It selects one with the minimum routing cost among relay candidate nodes. The routing protocol in the selected relay node provides the MAC layer with the address of the next hop node on the path to the sink node. Then, the MAC layer in the selected relay node forwards the erroneous data packet to the next hop node rather than a receiver node. Performance studies are carried out through simulation. Simulation results show that the proposed protocol has about 21.8% lower average delay and about 14.4% lower average number of nodes passed than the previous protocol, regardless of the maximum transmission range.

Keywords: cooperative communication, cross layer, MAC, routing, USN

1. Introduction

Underwater sensor networks (USNs) are one of types of sensor networks used in underwater environments [1]. USNs are characterized by low available bandwidth, long propagation delay, and high bit errors. Due to these characteristics, medium access control (MAC) protocols proposed for wireless networks cannot be used directly in USNs. Therefore, it is necessary to design new MAC protocols in consideration of the characteristics of the USNs.

Several MAC protocols for USNs have been proposed to address the disadvantages of the harsh underwater environments [2-6]. Most of them are based on the Request-to-Send (RTS) and Clear-to-Send (CTS) handshaking in order to reduce the packet collisions [7]. In Reference [2], the authors proposed a multiple access collision avoidance protocol for underwater (MACA-U) with the applied terrestrial MACA protocol. In the MACA-U protocol, a sender node transmits a RTS packet to a receiver node after channel contention. After receiving the RTS packet, the receiver node transmits a CTS packet. Then, the sender node transmits its own data packet to the receiver node.

The handshaking-based MAC protocols cause the low channel utilization due to the presence of long propagation delays, which in turn severely affects network performance. Cooperative communications have been introduced to improve the overall performance with the help of relay nodes [8-13]. The cooperative communication is much faster than the direct communication. The direct communication means sending data packets directly to a receiver node.

∗ Corresponding author. E-mail address: sunmyeng@kumoh.ac.kr

Tel.: +82-54-478-7547; Fax: +82-54-478-7539

To apply the cooperative communication to USNs, several protocols have been proposed [14-18]. Underwater cooperative MAC (UCMAC) was proposed in [14]. A sender node provides a receiver node with a list of relay nodes. For erroneous reception of data packets, the receiver node requests retransmission to the relay nodes in a closest-one-first manner. A relay node transmits the buffered data packet to the receiver node. If no available relay nodes exist, the UCMAC protocol follows the MACA-U protocol. In order to increase the reliability, Tran-Dang, *et al*. proposed a CACR (Channel-Aware Cooperative Routing) protocol in [15]. The CACR protocol uses two relays: a routing relay (RR) for forwarding data packets on routing paths, and a cooperative relay (CR) for one-hop cooperative communications. Before sending a data packet, a sender node selects both RR and CR nodes based on the link quality indicators, such as signal-to-noise ratio (SNR), time of arrival, and hop count to the sink node. Then, the sender node transmits a data packet to the RR node by the broadcast nature of the transmission medium. The CR node transmits the overheard data packet to the RR node separately. The RR node combines the signals received from the sender node and the CR node by the diversity-combining techniques. Reference [16] proposed a 2-hop routing protocol called the channel aware energy efficient two-hop cooperative routing protocol (CEETHCoR) to minimize the transmission energy. A sender node selects three nodes based on the link quality indicators: a next-hop node, a relay node, and a next two-hop node. The sender node transmits a data packet to a next two-hop node via a next-hop node and a relay node. The next two-hop node performs decoding the two signals received from the next-hop node and the relay node.

The legacy cooperative protocols select relay nodes to forward data packets on a hop-by-hop basis. Each node has a different path to the sink. Relay nodes always forward data packets to receiver nodes even though the receiver nodes are not on the best path. As a result, traditional hop-by-hop based cooperative protocols have limitations in improving performance. To improve network performance, this paper proposes a new cross layer based cooperative communication (CLCC) protocol. The CLCC protocol involves two layers: MAC layer and Network layer. The proposed protocol chooses the best relay node based on the information provided by a routing protocol at the network layer. The routing protocol in the selected relay node provides the MAC layer with the address of the next hop node on the path to the sink node. Then, the MAC layer in the selected relay node forwards the data packet to the next hop node rather than the receiver node.

This paper is organized as follows. Section 2 briefly describes related work. Section 3 describes the proposed CLCC protocol in detail. In Section 4, performance studies are carried out through simulation results. Finally, conclusions are drawn in Section 5.

2. Related Work

There have been a large number of studies focusing on MAC protocol in USNs. This section summarizes the CoopMAC, MACA-U, and UCMAC protocols. The CoopMAC protocol was proposed for the terrestrial wireless networks [8]. The MACA-U protocol was proposed for multi-hop USNs [2]. The UCMAC protocol was proposed based on cooperative communication for USNs [14].

2.1. CoopMAC protocol

Every node has a CoopTable. After receiving RTS, CTS, DATA, or ACK from a neighbor node, a node updates its own CoopTable. Table 1 shows the CoopTable format. The CoopTable contains four fields. The first field is a MAC address of a helper candidate. Here, a helper means a relay. The second field is the time that a node last received a packet from the candidate. The third field is the data rate between a helper candidate and a receiver node. The last field is the data rate between a sender node and a helper candidate. When neighbor nodes transmit a packet, a node overhears the packet, estimates the data rate, and updates its own CoopTable. If the estimated data rate meets the following condition, the node stores the sender node of the packet in the CoopTable.

$$1/R_{S,H} + 1/R_{H,D} > 1/R_{S,D} \tag{1}$$

where, $R_{S,D}$, $R_{S,H}$, and $R_{H,D}$ denote data rate between a sender node (S) and a receiver node (D), that between a sender node (S) and a helper node (H), and that between a helper node (H) and a receiver node (D), respectively.

Table 1 CoopTable format

Address	Time	$R_{H,D}$	$R_{S,H}$
H_1	T_1	$R_{H1,D}$	$R_{S,H1}$
...	...	...	...
H_n	Tn	$R_{Hn,D}$	$R_{S,Hn}$

When a sender node has data packets to transmit, it searches the CoopTable to find helper node candidates. If it finds more than one candidate, it will calculate the packet transmission time of each candidate and select the node with the shortest one as a helper node.

After selecting a helper node, a sender node transmits a CoopRTS packet to the selected node. After receiving the CoopRTS packet, the helper node sends a Helper ready To Send (HTS) packet to the sender node and the receiver node. After receiving the HTS packet from the selected helper node, the receiver node transmits a CTS packet to the sender node. Finally, the sender node sends a data packet to the helper node which forwards the packet to the receiver node.

Fig. 1 shows the procedure of the CoopMAC protocol in brief. Fig. 1(a) indicates a control packet exchange, and Fig. 1(b) presents a data packet exchange.

(a) Control packet exchange   (b) Data packet exchange

Fig. 1 Procedure of the CoopMAC protocol

2.2. MACA-U protocol

In the MACA-U protocol, when a sender node has a data packet to send, it uses a handshaking procedure. That is, a sender node transmits a RTS packet to a receiver node after channel contention. After receiving the RTS packet, the receiver node transmits a CTS packet. When other nodes receive the RTS or CTS packets, they set their timer and do not participate in the data packet transmission process. After receiving the CTS packet from the receiver node, the sender node transmits its own data packet to the receiver node. Finally, the receiver node sends an ACK packet to the sender node.

Fig. 2 shows an operation of MACA-U protocol. Fig. 2(a) shows the procedure when no error occurs in the data packet. There are four nodes (S, N1, N2, and D). S and D are a sender node and a receiver node, respectively. N1 and N2 are neighbor nodes of the sender node S. When the sender node S has a data packet to send, it starts its backoff procedure. If its backoff counter reaches zero, it broadcasts a RTS packet. The receiver 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.

Fig. 2(b) illustrates the procedure when an error occurs in the data packet. The sender nodes S has a data packet to send. After the backoff procedure, the node S broadcasts a RTS packet. It receives a CTS packet from the receiver node D and transmits the DATA packet. The node D does not do anything since an error occurs in the data packet. The timer for the data packet expires, and the node S repeats the backoff procedure. Fig. 2 (b) shows that if an error occurs in the data packet, it takes more time to successfully deliver the data packet to the receiver node.
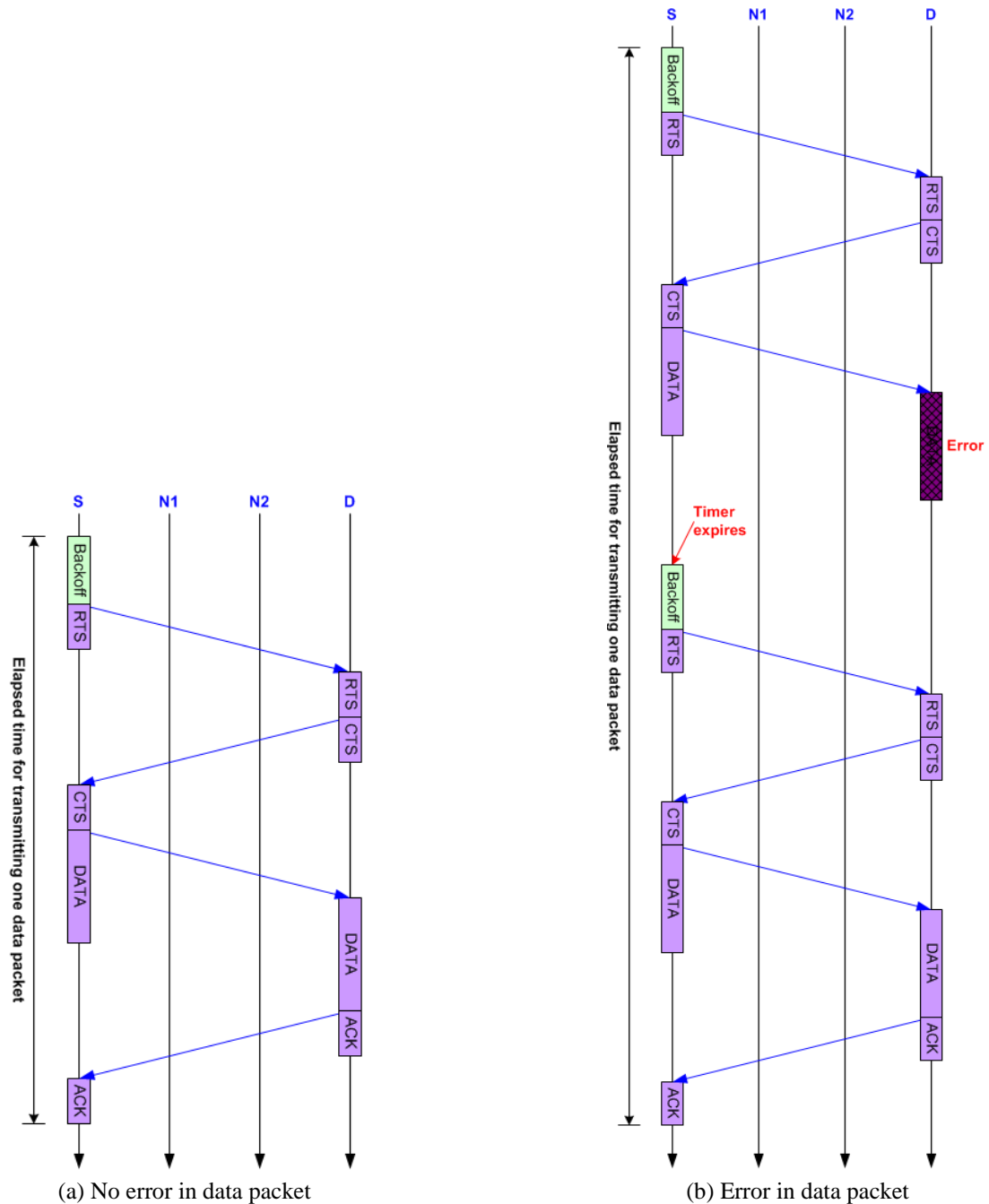
(a) No error in data packet   (b) Error in data packet

Fig. 2 Operation of MACA-U protocol

2.3. UCMAC protocol

The UCMAC protocol is based on cooperative communications. The operation of the UCMAC protocol is similar to that of the MACA-U protocol except for erroneous data packets. In the MACA-U protocol, if a receiver node receives an erroneous data packet, it does not do anything. However, in the UCMAC protocol, the receiver requests retransmission of the data packet to relay nodes.

Fig. 3 shows an operation of UCMAC protocol. A sender node S transmits a RTS packet including the propagation delay between itself and a receiver node D after the channel contention. Overhearing the RTS packet, a neighbor node N determines whether it is an eligible relay node based on the following criteria:

$$PD_{S,N} < PD_{S,D} \text{ and } PD_{N,D} < PD_{S,D} \tag{2}$$

where, $PD_{x,y}$ means the propagation delay between nodes x and y.

A neighbor node satisfying (2) sends a request-to-cooperate (RTC) packet including the propagation delay between itself and a receiver node. In Fig. 3, neighbor nodes N1 and N2 send the RTC packets to the sender node S.

The receiver node sends a CTS packet. After receiving RTC packets from its neighbor nodes and a CTS packet from the receiver node, the sender node makes a list of relay nodes based on the propagation delays in the RTC packets. Then, it transmits a data packet including the relay node list. The nodes on the list save the data packet after receiving it. For erroneous reception of the data packet, the receiver node selects a relay node with the shortest propagation delay from the relay node list. Subsequently, it sends a NACK packet to the relay node to request retransmission of the data packet. The relay node transmits a stored data packet. Once the retransmission is successful, the receiver node sends ACK to the sender node. Overhearing the ACK, the nodes on the list recognize the completion and discard the stored data packet.

Figs. 2 and 3 indicate that the elapsed time for transmitting one data packet in the UCMAC protocol is much shorter than that in the MACA-U protocol.
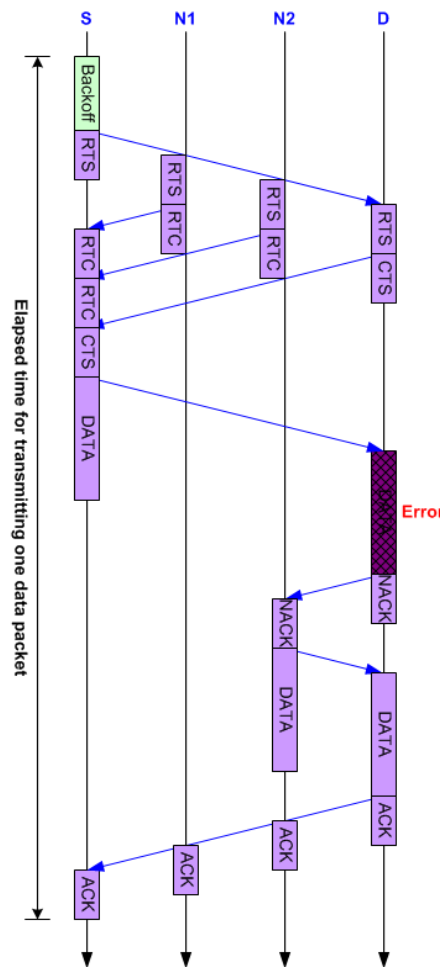

Fig. 3 Operation of UCMAC protocol (Error in data packet)

3. CLCC Protocol

Here is an example to illustrate the basic operation of the proposed CLCC protocol (refer to Fig. 4. Fig. 4(a) presents a topology with five sensor nodes (N1~N5) and one sink node. The topology is modeled as a weighted graph. Each node represents a sensor node, and each edge stands for a physical link. The edge labels specify the routing cost of the link. Because the routing protocol is out of the scope of this paper, it is not explained how to get the cost in this study.

Figs. 4(b)-4(c) show a routing path and a routing path table from each sensor node to the sink node, respectively. A sensor node chooses a path to the sink node in such a way that the total cost for the path is the least cost among all possible paths. In

Fig. 4(c), the field cost is the sum of the edge costs along the path. Next hop is a routing term that refers to the next nearest node in the routing path through which a packet can pass.

Fig. 4(d) indicates an example of hop-by-hop cooperative communication. The sensor node 1 transmits a data packet to the next hop node (N4) on the routing path. The data packet includes a list of relay candidate nodes (N2 and N3). After receiving the data packet, N2 and N3 store it in their buffer to prepare for the retransmission request from N4. If N4 does not successfully receive the data packet, it selects the closest sensor node (i.e., N2) as a relay node and requests retransmission to the selected relay node. The relay node N2 forwards the data packet to N4.

Fig. 4(e) illustrates an example of the proposed cross layer cooperative communication. The proposed protocol chooses a relay node based on the routing cost provided by a routing protocol at the network layer. For erroneous reception of the data packet, N4 selects N3 with the minimum routing cost as a relay node and requests retransmission to the relay node. N3 transmits the data packet to the next hop N5 on the routing path to the sink node rather than N4.
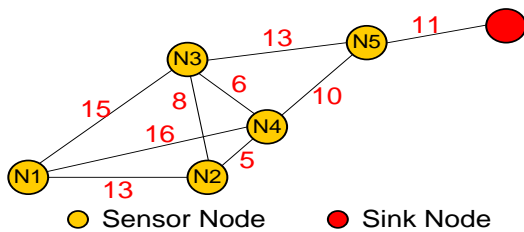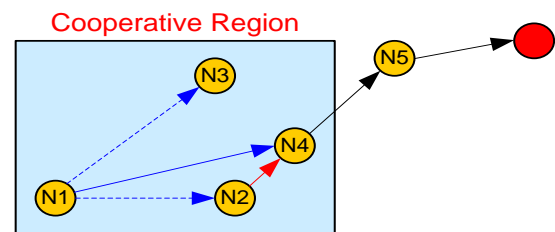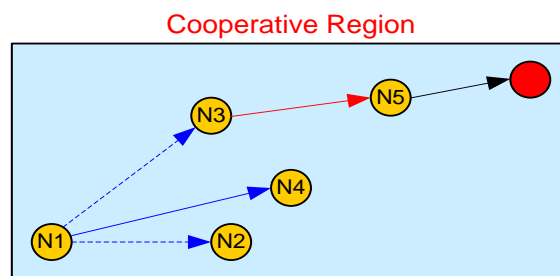

(a) Example topology
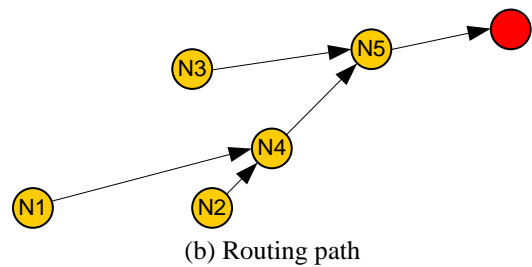
(b) Routing path

Node	Cost	Next Hop	Path
N1	37	N4	N1 - N4 - N5 - Sink
N2	26	N4	N2 - N4 - N5 - Sink
N3	24	N5	N3 - N5 - Sink
N4	21	N5	N4 - N5 - Sink
N5	11	Sink	N5 - Sink

(c) Routing path table

(d) Hop by hop cooperative communication

(e) Cross layer cooperative communication

Fig. 4 Example of cooperative communication

The proposed protocol defines the cooperative region as the area that is considered to obtain parameter values used when selecting a relay node in cooperative communication. The cooperative region in the hop by hop cooperative communication is limited (i.e., single hop of a sender node) (refer to Fig. 4(d)). The hop by hop cooperative communication focuses only on how to quickly send data packets to the receiver node within a single hop. However, the cooperative region in the cross layer cooperative communication is very large (i.e., network wide) (refer to Fig. 4(e)). The cross layer cooperative communication focuses on how to quickly send data packets to the sink node. Therefore, the cross layer cooperative communication can provide better network performance.

Fig. 5 shows the cross layer architecture of the proposed CLCC protocol. By sharing data, the cross layer principle can be extended to the routing and the cooperative communication. The network status is responsible for storing the information
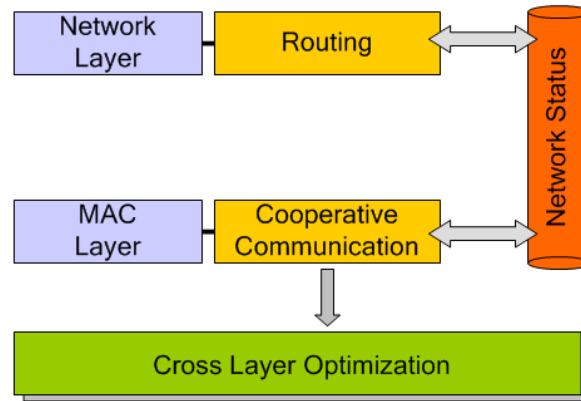
Fig. 5 Cross layer architecture of the proposed CLCC protocol

collected by the protocol through the stack structure. Each protocol can access the network status to share data with other protocols.

The main focus of this paper is to design an efficient cooperative communication protocol that transmits data packets from a sensor node to the sink node. Routing protocols are out of the scope of this paper. Therefore, the proposed protocol assumes that the path from a sensor node to the sink node is already established.

The routing component provides routing cost information to the network status. The cooperative communication component reads the information from the network status.

For erroneous reception of a data packet, a receiver node selects one with the minimum routing cost among the relay candidate nodes as a relay node and requests retransmission to the selected relay node. The selected relay node notifies the routing protocol of its selection as a relay node through the network status instead of immediately retransmitting the data packet. The routing protocol component looks up the routing table and provides the address of next hop node on the path to the sink node to the cooperative communication component through the network status component. After receiving the address of the next hop node, the cooperative communication component compares the address to the receiver node. If the comparison result is the same, the relay node forwards the data packet to the receiver node. Otherwise, it transmits an ACK packet to the sender node and the receiver node, then sends the data packet to the next hop node.

Fig. 6 shows an operation of the proposed CLCC protocol based on the topology in Fig. 4. When the sender node N1 has a data packet to send, it starts its backoff procedure. If its backoff counter reaches zero, it broadcasts a RTS packet. The receiver node N4 sends a CTS packet after receiving the RTS packet. Afterwards, N1 transmits its data packet. The neighbor nodes N2 and N3 save the data packet after overhearing it. For erroneous reception of the data packet, N4 selects a relay node with the minimum routing cost. To explain Fig. 6(a), assume that N2 has lower routing cost than N3 in Fig. 4. N4 selects N2 as a relay node and sends NACK packet to N2 to request retransmission of the data packet. The routing protocol component in N2 looks up the routing table and provides the cooperative communication component with the address of the next hop node (N4) on the path to the sink node through the network status component. After receiving the address of the next hop node, the cooperative communication component compares the address (N4) to the receiver node (N4). N2 forwards the data packet to N4 since the addresses are same.

To explain Fig. 6(b), assume that N3 has lower routing cost than N2 in Fig. 4. N4 selects N3 as a relay node and sends NACK packet to N3 to request retransmission of the data packet. The routing protocol component in N3 looks up the routing table and provides the address of the next hop node (N5) on the path to the cooperative communication component through the network status component. After receiving the address of the next hop node, the cooperative communication component compares the address (N5) to the receiver node (N4). N3 sends ACK to N1 and N4, then forwards the data packet to N5 rather than N4 since the addresses are same. Finally, N5 broadcasts ACK to N3.
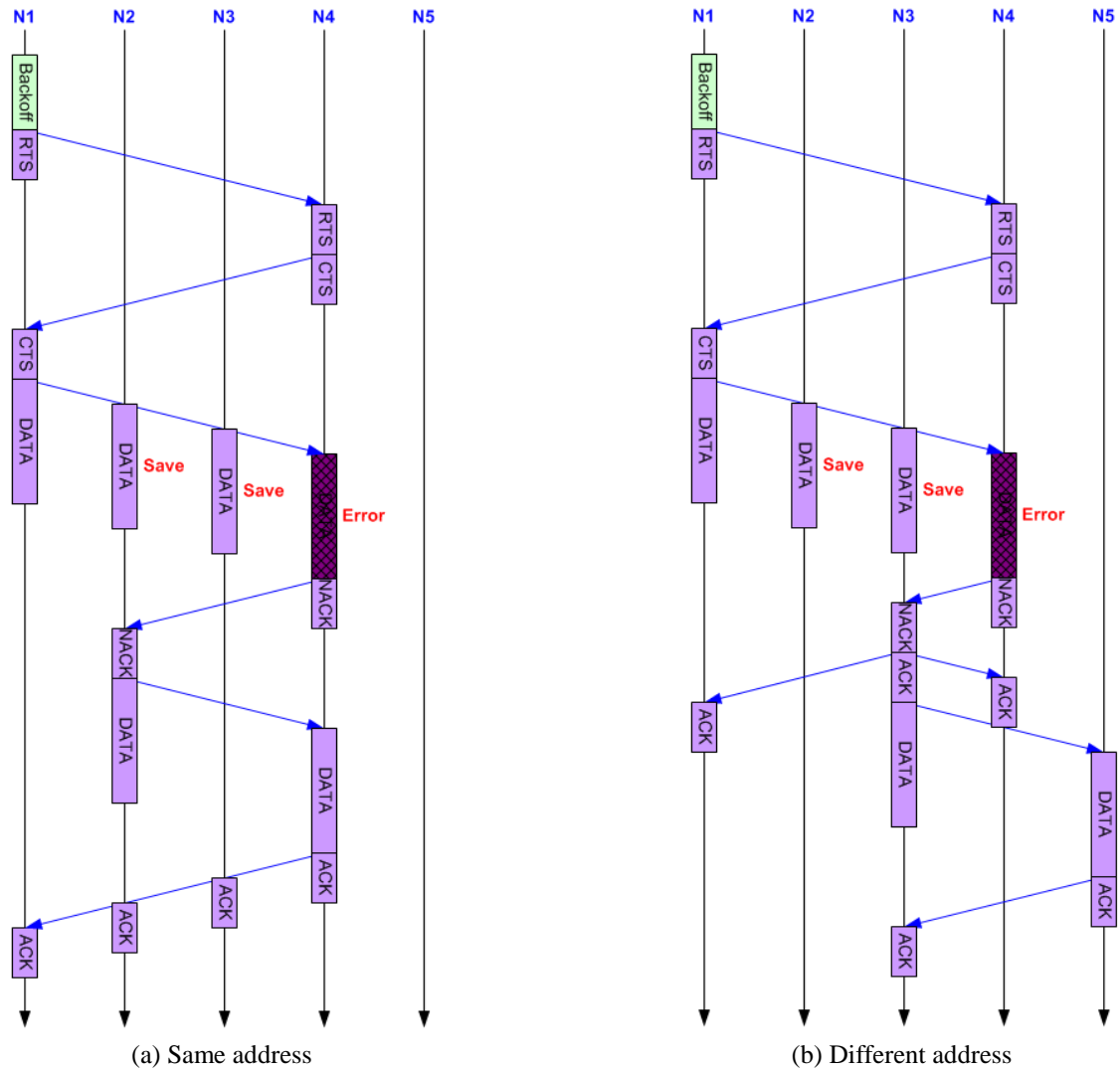
(a) Same address   (b) Different address

Fig. 6 Operation of proposed CLCC protocol

4. Simulation Results

This section analyzes simulation results of the proposed CLCC protocol. The proposed CLCC protocol was implemented in C++. The performance of the CLCC protocol is compared with that of the UCMAC protocol.
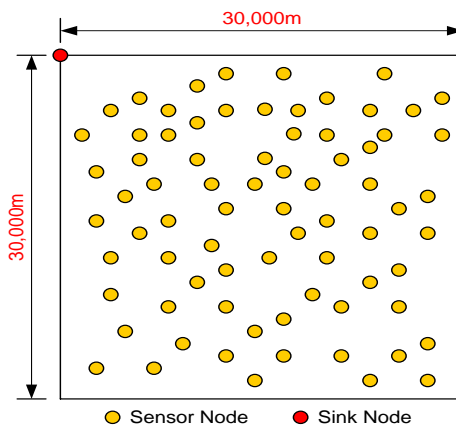

Fig. 7 Simulation topology

The simulation topology is shown in Fig. 7. In the topology, there are many sensor nodes and one sink node. Sensor nodes have data packets to send to the sink node. Sensor nodes are randomly distributed in a square area of 30,000m × 30,000m. The sink node is located at the top left (0, 0).

The system parameters used in the simulation are listed in Table 2. The maximum data rate is 4,800 bps; the length of control packets, such as RTS, CTS, ACK, and NACK, is 256 bits; a constant data packet size of 8,192 bits is used; the sound speed is 1500m/s.

Table 2 Simulation parameters

Parameter	Value
RTS	256 bits
CTS	256 bits
ACK	256 bits
NACK	256 bits
DATA	8,192 bits
Data Rate	4,800 bps
Sound Speed	1,500 m/s

The main performance metric of interest are average delay and the number of passed nodes. The average delay is the time between a data packet arrival at the queue of a sender node and the successful data packet transmission to the sink node. The number of passed nodes is what have passed until one data packet reaches the sink node. All simulation results were averaged over 10 simulations.

Figs. 8-9 demonstrate simulation results according to the maximum transmission range. In the figures, the maximum transmission range is increased by 100m from 800m to 1,600m. Fig. 8 shows the average delay and illustrates that the average delay decreases as the maximum transmission range becomes larger. The proposed CLCC protocol always outperforms the UCMAC protocol; the UCMAC protocol operates on a hop by hop basis. Therefore, it just selects the best relay node in a single hop. Nevertheless, the CLCC protocol considers the cross layer principle. Accordingly, it selects the best relay node based on routing cost through the network.
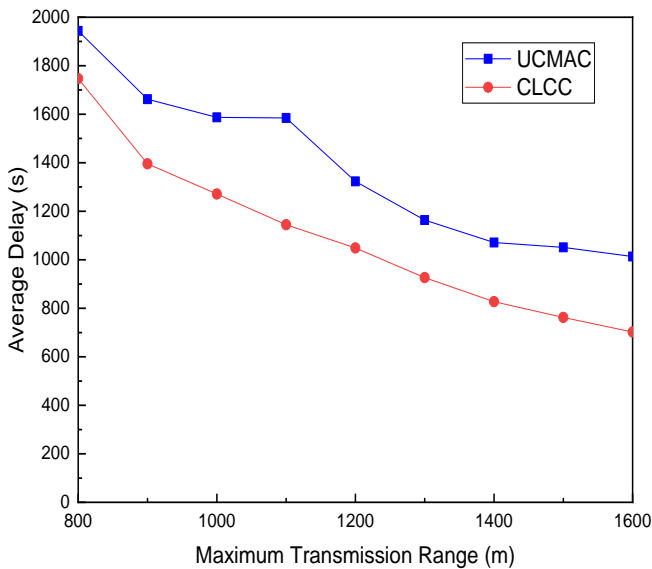

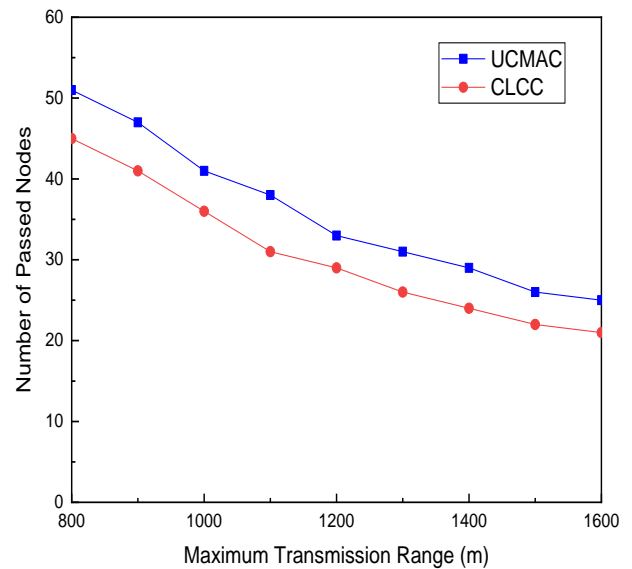Fig. 8 Average delay according to the maximum transmission range

Fig. 9 Number of passed nodes according to the maximum transmission range

Fig. 9 shows the effect of the transmission range on the number of passed nodes. As the maximum transmission range gets larger, the number of passed nodes is decreasing. A sensor node chooses a more faraway node as a relay node when the transmission range grows. Thus, the number of passed nodes is getting smaller. The proposed CLCC protocol always has lower number of nodes passed than the UCMAC protocol, regardless of the maximum transmission range.

Figs. 10-11 demonstrate the performance results according to the packet error rate. In the figures, the packet error rate is increased by 5% from 0% to 30%. Fig. 10 depicts the average delay according to the packet error rate. As the packet error rate increases, the probability that a receiver node receives erroneous data packets increases. Accordingly, a receiver node usually

requests retransmissions of the data packets. For the UCMAC protocol, the average delay increases almost linearly as the packet error rate is getting larger. However, the proposed CLCC protocol grows more slowly than the UCMAC protocol.

Fig. 11 depicts the simulation results for the number of passed nodes. As the packet error rate gets larger, the number of passed nodes is increasing. A receiver node usually chooses relay nodes when the packet error rate grows. Therefore, the number of nodes passed is getting larger. The proposed CLCC protocol always has lower number of nodes passed than the UCMAC protocol, regardless of the packet error rate.
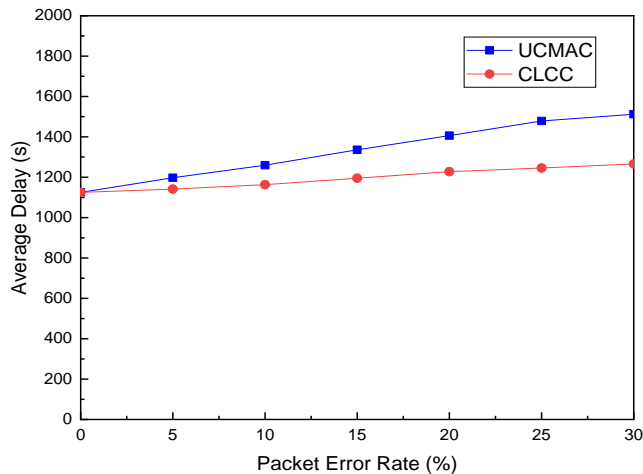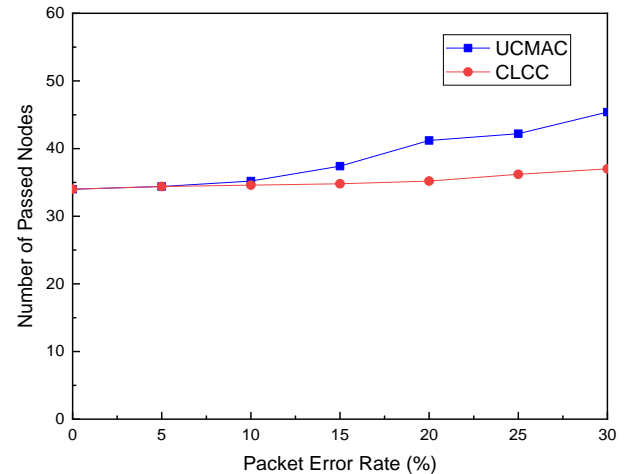
Fig. 10 Average delay according to the packet error rate

Fig. 11 Number of passed nodes according to the packet error rate

5. Conclusions

Since USNs have a long propagation delay in exchanging control and data packets, handshaking-based MAC protocols are not suitable for underwater environments. In order to provide better network performance, cooperative communications have been introduced. Legacy cooperative communication protocols select relay nodes on a hop-by-hop basis. Therefore, they have limitations in improving performance. To get better network performance, a new cross layer based cooperative communication protocol was proposed in this paper. The proposed protocol considered the interaction between the MAC layer and the network layer. The proposed protocol chooses a relay node based on the routing cost provided by a routing protocol at the network layer. The selected relay node forwards a data packet to the next hop node on the path to the sink node. Simulation results show that the proposed protocol is much better than the previous protocol in terms of average delay and number of passed nodes.

This study has several limitations. The proposed protocol involves two layers: MAC layer and Network layer. The MAC layer selects a best relay node based on the routing information provided by the network layer. The MAC layer information is not provided to the network layer. Thus, there is a limit in establishing the best path to the sink node at the network layer, which means the lack of interaction between two layers. To select a better relay node at the MAC layer and to establish a better path to the sink node at the network layer, it needs to increase the degree of interaction between two layers. In addition, the proposed protocol does not utilize the channel quality indicators, such as received signal strength indication (RSSI) and signal-to-noise ratio (SNR) obtained from the physical layer. The physical layer information should be used to cope with variations in the channel conditions. In order to provide optimal performance, it is necessary to research cross layer cooperative communication protocols including physical, MAC, and network layers.

Acknowledgments

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] P. Casari and M. Zorzi, "Protocol design issues in underwater acoustic networks," Computer Communications, vol. 34, no. 17, pp. 2013-2025, November 2011.

[2] H. H. Ng, W. S. Soh, and M. Motani, "MACA-U: a media access protocol for underwater acoustic networks," Proc. IEEE Global Communications Conference (GLOBECOM 08), IEEE Press, December 2008, pp. 1-5.

[3] F. A. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "An energy-conserving collision-free MAC protocol for underwater sensor networks," IEEE Access, vol. 7, pp. 27155-27171, February 2019.

[4] Z. Xi, X. Kan, L. Cao, H. Liu, G. Manogaran, G. Mastorakis, and C. X. Mavromoustakis, "Research on underwater wireless sensor network and MAC protocol and location algorithm," IEEE Access, vol. 7, pp. 56606-56616, February 2019.

[5] Z. Zhang, W. Shi, Q. Niu, Y. Guo, J. Wang, and H. Luo, "A load-based hybrid MAC protocol for underwater wireless sensor networks," IEEE Access, vol. 7, pp. 104542-104552, July 2019.

[6] X. Feng, Z. Wang, G. Han, W. Qu, and A. Chen, "Distributed receiver-oriented adaptive multichannel MAC for underwater sensor networks," IEEE Access, vol. 6, pp. 11666-11675, February 2018.

[7] S. Kim, "Dual polling protocol for improving performance in wireless Ad Hoc networks," International Journal of Engineering and Technology Innovation, vol. 8, no. 1, pp. 1-12, January 2018.

[8] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, "CoopMAC: a cooperative MAC for wireless LANs," IEEE Journal on Selected Areas in Communications, vol. 25, no. 2, pp. 340-354, February 2007.

[9] Y. Su, X. Lu, Y. Zhao, L. Huang, and X. Du, "Cooperative communications with relay selection based on deep reinforcement learning in wireless sensor networks," IEEE Sensors Journal, vol. 19, no.20, pp. 9561-9569, October 2019.

[10] A. S. Shah, H. Ilhan, and U. Tureli, "RECV-MAC: a novel reliable and efficient cooperative MAC protocol for VANETs," IET Communications, vol. 13, no. 16, pp. 2541-2549, October 2019.

[11] H. Zhu and G. Cao, "rDCF: a relay-enabled medium access control protocol for wireless ad hoc networks," IEEE Transactions on Mobile Computing, vol. 5, no. 9, pp. 1201-1214, Septptember 2006.

[12] L. Guo, X. Ding, H. Wang, Q. Li, S. Chen, and X. Zhang, "Cooperative relay service in a wireless LAN," IEEE Journal on Selected Areas in Communications, vol. 25, no. 2, pp. 355-368, February 2007.

[13] B. Zhang and X. Jia, "Multi-hop collaborative relay networks with consideration of contention overhead of relay nodes in IEEE 802.11 DCF," IEEE Transactions on Communications, vol. 61, no. 2, pp. 532-540, February 2013.

[14] H. W. Kim, T. H. Im, and H. S. Cho, "UCMAC: a cooperative MAC protocol for underwater wireless sensor networks," Sensors, vol. 18, no. 6, June 2018.

[15] H. Tran-Dang and D. S. Kim, "Channel-aware cooperative routing in underwater acoustic sensor networks," Journal of Communications and Networks, vol. 21, no. 1, pp. 33-44, February 2019.

[16] H. Tran-Dang and D. S. Kim, "Channel-aware energy-efficient two-hop cooperative routing protocol for underwater acoustic sensor networks," IEEE Access, vol. 7, pp. 63181-63194, May 2019.

[17] S. Ahmed, N. Javaid, F. A. Khan, M. Y. Durrani, A. Ali, A. Shaukat, and U. Qasim, et al., "Co-UWSN: cooperative energy-efficient protocol for underwater WSNs," International Journal of Distributed Sensor Networks, vol. 11, no. 4, April 2015.

[18] S. Al-Dharrab, A. Muqaibel, and M. Uysal, "Performance of multicarrier cooperative communication systems over underwater acoustic channels," IET Communications, vol. 11, no. 12, pp. 1941-1951, Auguest 2017.