



CUMAC-CAM: a channel allocation aware MAC protocol for addressing triple hidden terminal problems in multi-channel UWSNs



Purobi Rahman¹ · Amit Karmaker¹ · Mohammad Shah Alam^{1,2}  · Mohammad Asadul Hoque³ · William L. Lambert²

© Springer Nature Switzerland AG 2019

Abstract

In this paper, a cooperative underwater multi-channel MAC (CUMAC) protocol has been proposed with both delay mapping and channel allocation assessment in order to improve network performance and handle triple hidden terminal (THT) problems in underwater sensor networks. A novel channel allocation matrix (CAM) was developed for estimating propagation delay and increasing utilization of channel. In the proposed scheme, every node maintains a database for delay mapping, based on which the sender runs a scheduling algorithm prior to transmitting any data. This delay mapping database assists a node in predicting packet collision probability. The overall objectives are—first, to increase the rate of successful transmission through mitigation of THT problems in multi-channel underwater sensor networks; and second, to increase channel utilization leveraging the database of delay mapping and channel allocation assessment. Results from performance evaluation demonstrate the efficiency of the proposed CUMAC-CAM protocol in terms of packet delivery ratio, energy consumption, end-to-end delay, network throughput, collision probability, packet loss ratio and fairness index compared to the contemporary CUMAC protocol and RTS/CTS based multi-channel MAC protocols.

Keywords UWSN · Multi-channel hidden terminal problem · Channel allocation matrix (CAM) · Transmission scheduling

1 Introduction

Acoustic communication protocols, similar to Radio Frequency (RF) communications, have started implementing the concept of multiple channels utilization in underwater sensor networks (UWSN). Multi-channel communication increases throughput significantly. However, it also induces collision probability for hidden terminal problem. Specifically, the Triple Hidden Terminal (THT) problem significantly exists in underwater sensor networks. It occurs due to collision occurrence which elicits from multi-hop and multichannel communication accompanied by long propagation delay [1–4].

Prevailing MAC protocols attempt to minimize the detrimental THT impact without considering the facts

obtained from propagation delay, which could be leveraged to significantly improve the performance of underwater sensor networks. Hence, this research suggests a protocol—cooperative underwater multi-channel MAC protocol along with delay mapping and channel allotment assessment. A novel channel allocation matrix (CAM) has been presented with an aim of assessing propagation delay to ensure effective channel use. In CUMAC, every node keeps a database for delay mapping. Accordingly, sender or receiver executes a scheduling algorithm based upon this delay mapping information before originating any transmission. This delay mapping information helps a node in predicting the probability of packet collision with other nodes transmissions. In summary, the aim is to make sure of successful

✉ Mohammad Shah Alam, msalam@tntech.edu; Purobi Rahman, purobi.rahman@gmail.com; Amit Karmaker, amitkarmaker06@gmail.com; Mohammad Asadul Hoque, hoquem@etsu.edu; William L. Lambert, wlambert42@students.tntech.edu | ¹IICT, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh. ²Department of Computer Science, Tennessee Technological University, Cookeville, USA. ³Department of Computing, East Tennessee State University, Johnson City, USA.

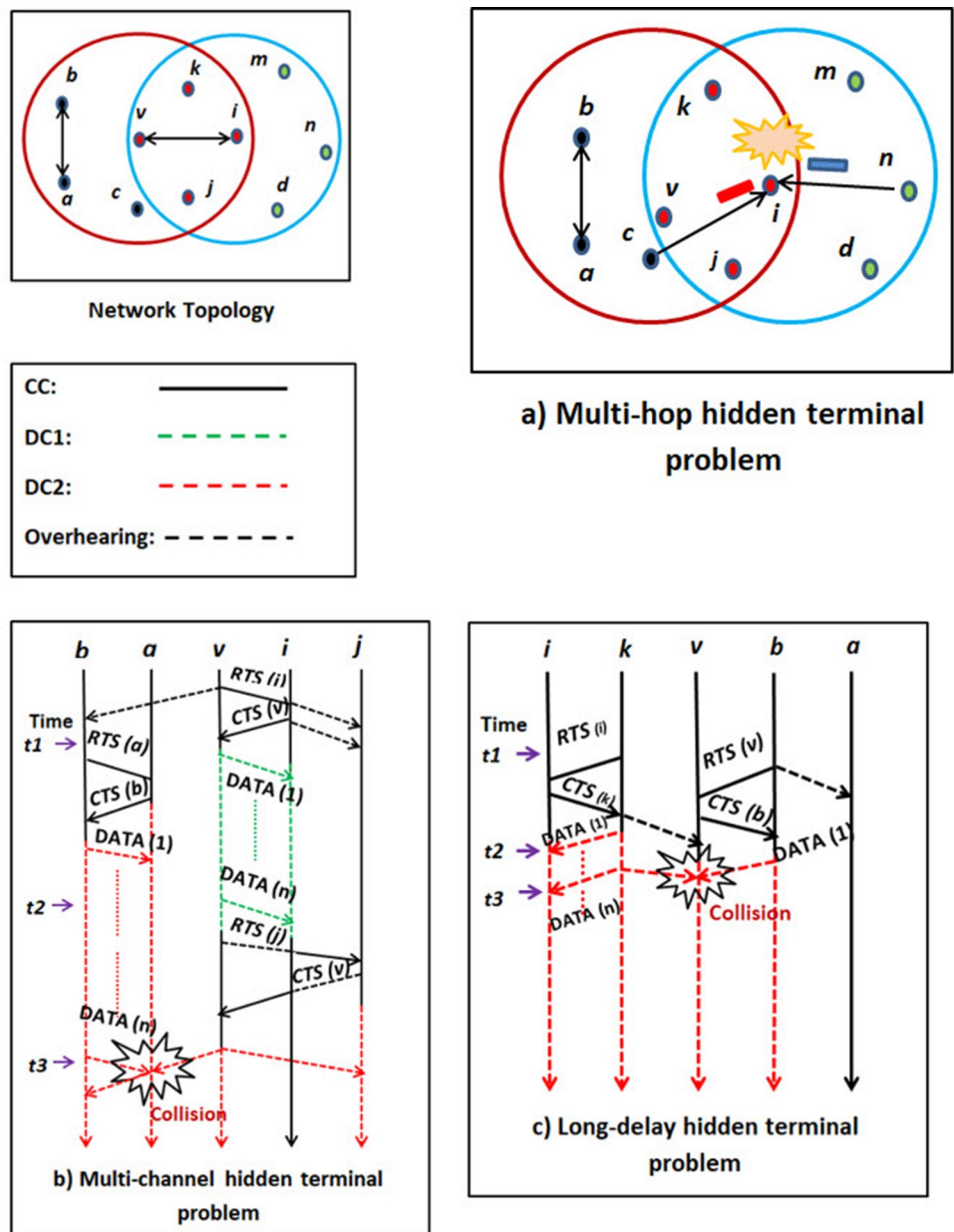


transmission by alleviating problems in multichannel UWSNs, in addition to enhancing the channel utilization through effective use of delay mapping information and channel allocation analysis. Performance evaluation through simulation analysis demonstrates that proposed CUMAC-CAM protocol is more effective in respect of network throughput, energy consumption, end to end delay, collision probability, packet delivery ratio, packet loss ratio, and fairness index, when compared to the contemporary cooperative underwater multichannel MAC (CUMAC) protocol and those based on RTS/CTS scheme.

Figure 1 shows a network topology with number of nodes $a, b, c, d, v, k, i, j, m, n$ communicating over multiple

channels, including a CC (control channel) and two DCs (data channels). Figure 1a depicts a scenario of a typical hidden terminal problem with multi-hop communication. According to Fig. 1b, node v sends RTS to i over CC when v has some data for i . RTS consists of probable data channels and reservation information. Both two nodes move to the preferred data channel (DC1) at time t_1 then continue data transmission after RTS/CTS handshaking. Suppose, b has data for a for the period of (t_1, t_2) . Therefore, node b and node a both move to the other idle data channel (DC2), after carrying out the reservation step. Node v and node i do not overhear control channel (CC) throughout the duration of t_1, t_2 as they are continuing

Fig. 1 Triple hidden terminal (THT) problem



their transmission over that period. Henceforth, node v and i presume that the data channel DC2 is still in an idle state. At time T3, packet collision happens at node a or b for one situation: when v completes data transmission to i , and v has data for transmitting to j . At that time, if v also wants to go for DC2 channel, which is already occupied by a and b , then collision occurs as a consequence of the multi-channel hidden terminal problem. In the same way, Fig. 1c portrays the long-delay hidden terminal problem. As illustrated in figure, node k selects channel DC2 after handshaking with node i . Later, node b and v also negotiate over CC for their upcoming transmission. Suppose, node v got the CTS of node k after it completes the data channel selection and send CTS back to node b . Suppose node v selects the same data channel DC2. As a result, collision occurs, as node v is not aware of DC2 reservation by node k . Therefore, this delay associated hidden terminal problem is referred to as the “long-delay hidden terminal problem”. Both CUMAC and previous studies on underwater network have paid attention to overcoming the challenges of the acoustic channel. However, very few works exploit the characteristics of large propagation delay as a means to increase throughput. This research enhances the previous work on CUMAC by integrating both propagation delay mapping and channel allocation assessment. CAM is introduced with an aim to improve channel utilization. Moreover, the propagation delay map database and CAM will be aligned to increase transmission success by eliminating THT problems in UWSNs. In brief, the key concept of CUMAC-CAM is the prediction of upcoming transmission, in a scheme which facilitates data transmission without collisions.

A preliminary version of this work was presented in [5]. The remainder of this paper is organized as follows. In Sect. 2, we describe the preliminaries related to all concepts and provide a summary of the state of the art. Section 3 presents the limitations of the contemporary CUMAC protocol. Our proposed CUMAC-CAM protocol, an extension to the CUMAC protocol, is described in Sect. 4. Section 5 is dedicated to results and performance analysis. Finally we conclude the paper in Sect. 6.

2 Related work

UWSN came up to researchers as an interesting research area for the last few years due to the fact of ocean exploration improvement and fulfillment of numbers of underwater applications [6]. Unlike electromagnetic waves, underwater acoustic channels are characterized by a long propagation delay, high path loss, low data rates, and a limited bandwidth [7–9]. Furthermore, energy required to transmit data in acoustic channels is almost

100 times than the energy required to receive data [10, 11]. Therefore, the goal of designing an efficient MAC protocol for UWSN is primarily focused on minimizing the number of collisions, at the cost of heavy duty cycling of sensor nodes. For delay sensitive underwater sensor network applications, opportunistic based routing (OR) has been implemented with an aim of maximizing output, along with achieving the end to end latency requirement [12]. Likewise, a novel MAC protocol-DTMAC is proposed to improve the network throughput, as well as to overcome the challenges introduced by the long propagation delay in UWSNs [13]. A MAC protocol (DOTS) is presented in [14] which focused on minimizing the adverse effects of limited bandwidth and long propagation delay. It can effectively support collision free concurrent transmission over a single channel by exploiting the long propagation delay of acoustic channels. Two other MAC protocols: multi-channel access with ALOHA and multi-channel access with RTS/CTS on a dedicated control channel are discussed in [15, 16].

On the other hand, multi-channel transmission in UWSNs also compensates the adversarial effect of elongated propagation delay as well as low data rate of acoustic channels. However, utilizing multi-channel in underwater networks, that suffers from elongated propagation delay, significantly introduces hidden terminal problem [17]. Afterward, a MAC protocol is proposed in [18], referred as CUMAC, to alleviate the triple hidden terminal problems in underwater sensor networks. In CUMAC, a collision detection method is implemented along with a simple tone device by way of cooperation of neighbor nodes. Based upon slot reservation, another multi-channel MAC protocol-UMMAC is proposed in [19]. In UMMAC, length of slot duration is fixed which helps to avoid collision. Additionally, UMMAC combines channel reservation and negotiation in order to reduce the overhead of time and energy, as well as utilize multiple channels to improve the network performance. Some other studies on efficient MAC protocol design for underwater sensor networks are found in [20–25]. While majority of previous studies focused on single-channel communications, a new trend has taken place to make use of multiple channels in UWSNs in recent years [26, 27]. Several channel allocation [28] and channel assignment [29, 30] algorithms are widely used in multi-channel cognitive ad-hoc networks. L.T. Tan et al. proposed a non-overlapping and an overlapping channel assignment algorithm in order to maximize the throughput of both cognitive and ad-hoc networks. In the non-overlapping assignment, a particular channel is assigned to each secondary user (SU). It maximizes the throughput if the number of channels is sufficiently large. However, assigning a dedicated channel for each sensor node is not feasible for a

high number of nodes. On the other hand, in the case of an overlapping channel assignment, a set of channels are assigned to a particular set of SUs, which are neighbors to each other. To resolve the channel contention due to this overlapping assignment, the authors have proposed a MAC protocol. A dedicated control channel is used for contention resolution among the nodes. The authors have used random backoff to reduce the RTS/CTS collision. However, in each cycle, synchronization is needed among the neighbor nodes. In [28], the authors proposed a semi-distributed cooperative spectrum sharing (SDCSS), where each secondary user shares its sensing results with the neighboring nodes, through a dedicated control channel. For this dynamic spectrum sharing, authors have proposed a synchronized multi-channel MAC protocol. The time is divided into fixed sized cycles, and in each cycle there are four phases. Firstly, nodes are synchronized with each other through the control channel. Secondly, each node senses the spectrum on the assigned channel. Thirdly, each node shares the spectrum sensing information in the control channel, and the secondary user employs SDCSS in order to decide the channel status of available channel. Finally, SUs participate in contention (using p-persistent CSMA) and transmission (4-way handshaking). Although the above channel assignment and channel allocation schemes improve the performance significantly, it also has a stringent requirement of synchronization among the nodes. Such a requirement is viable in an ad-hoc network environment, but not achievable in underwater sensor networks due to its inherently long propagation delay.

Consequently, this research work focuses on developing a multi-channel MAC protocol that takes advantages of cooperative communication and prior propagation delay awareness.

3 CUMAC

In case of CUMAC, a node starts channel negotiation with the intended receiver, if it has a packet to send. The channel negotiation process comprises of RTS and CTS control messages [18]. On the other hand, the receiving node collaborates with neighbor nodes before sending CTS back to the sender node for channel selection and collision detection. Figure 2 illustrates the multi-channel hidden terminal problem of underwater sensor network.

When channel negotiation is going on over the control channel between node a and node b then node c and d are continuing their communication on data channel 2. Therefore, both node c and d are not aware of the allocation of data channel 1 by node a and node b. Afterward, when node c intends to send a data packet to node d, it

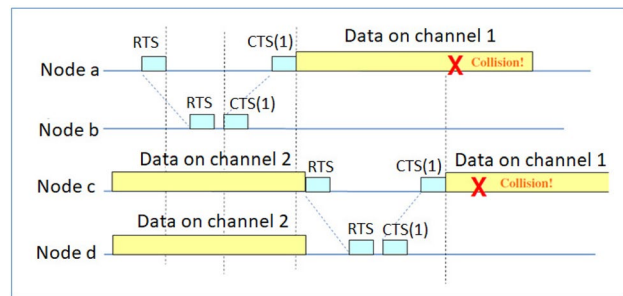


Fig. 2 Multichannel hidden terminal problem [18]

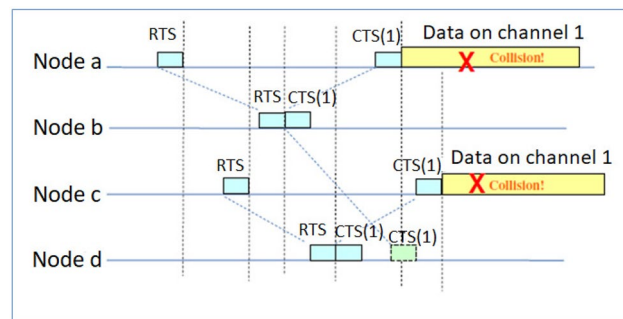


Fig. 3 Long delay hidden terminal problem [18]

then starts for handshaking over the control channel. As node d is not aware of the data channel 1 allocation by other nodes, it might select the same data channel. Hence, it increases the chances of collision occurrences.

The limitation of CUMAC, is that the sender node does not consider its neighbor node's on-going transmission and collision detection. Collision may also happen in the sender end, but CUMAC does not consider the sender side multi-channel hidden terminal problems for collision detection. Moreover, the long propagation delay hidden terminal problem has been addressed in CUMAC, which is depicted in Fig. 3.

However, it does not take advantage of the propagation delay information in order to both continue node communication and enhance channel utilization without any collision. In addition, every node maintains a channel usage table to keep records of data channel usage, but this channel usage table does not help receiver nodes to get the channel status for channel selection. Thus, the cooperative collision detection scheme asks for help from neighbor nodes. If a neighbor node sends a tone pulse sequence to the receiver node in order to notify on collision detection, then the receiver node selects another channel and it repeats the whole process of channel selection trials until it gets a correct channel.

4 Proposed protocol: CUMAC-CAM

The MAC protocols for UASN can be classified into two categories: centralized and distributed [31]. A centralized MAC protocol usually takes a long period of time in order to gather the necessary information (topology, transmission request, scheduling information, etc.) from sensor nodes, due to channel's long propagation delay [32]. Moreover, the failure of the central node causes the collapse of the whole network [26]. Therefore, distributed MAC protocols are usually more suitable, and are widely used in underwater sensor networks. In this paper, we also consider our proposed MAC protocol to work in distributed manner. This research work focuses on cooperative underwater multichannel MAC protocol (CUMAC) advancement by integrating propagation delay mapping along with channel allocation assessment. A channel allocation matrix (CAM) is introduced with an aim to enhance channel utilization. Moreover, both the propagation delay map database and CAM will be aligned for successful transmission by alleviating THT problems in underwater network. In brief, the key concept of the proposed scheme is the prediction of upcoming transmission, in such a way that facilitates data transmission that is free of collisions.

4.1 System model

- For simplicity, the underwater network considered here is a static underwater sensor network, meaning nodes in the network are static as well as uniformly distributed in a fixed area.
- Multi-hop topology of static nodes is considered and depicted in Fig. 4. The intermediary nodes of the network can communicate with each other in single hop distance in both directions.
- For each node, circular transmission range is considered as R and two nodes will not interfere with each other if their distance is larger than transmission range, R .
- The propagation speed of the acoustic signal is v and maximum propagation time T for acoustic signal from node to reach its transmission range is R/v .
- It is assumed that, while the network is initialized, the distance between nodes are calculated with the help of control packets by measuring round trip time or by sharing information between neighbor nodes.
- In keeping with assumption of CUMAC, there are multiple channels in the network. Each of the channels has equal bandwidth. Among the channels, there is one control channel and multiple data channels. Control channel is used for exchanging control messages and data channels for data transmission. Every node

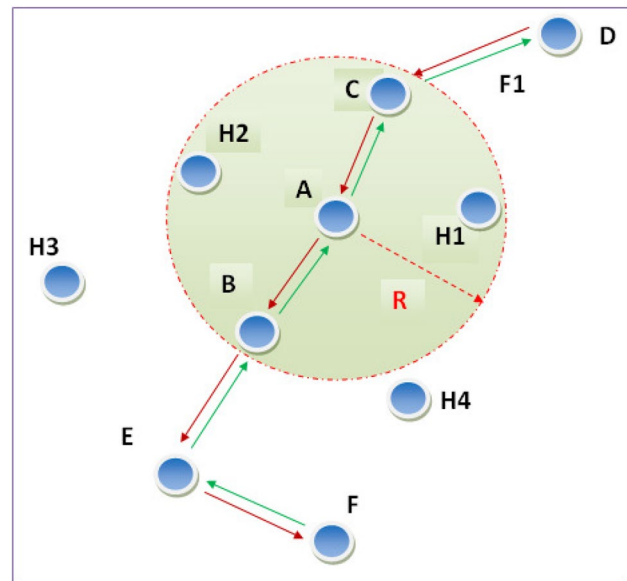


Fig. 4 Network topology

keeps listening to common control channel when there is nothing to send or receive.

- Every node has only one acoustic transceiver and thus node can dynamically switch to different channels whenever required. In addition, a node either works on control or data channel at once. That is, it cannot work on both types of channels at the same time.
- Proposed scheme suggests that each node in a network will contain a CAM along with delay map database, briefly explained in Sect. 4.2.1.
- A node will keep updating channel information into CAM and propagation delay into delay map database accordingly by passively overhearing RTS/CTS transmission over control channel.
- Packets arrives at nodes by following poisson process with arrival rate (λ packets per second over a small interval). Therefore, the packet generation of every node is poisson distributed.
- Furthermore, It is anticipated that, all nodes are aware of own location details through relevant localization algorithms which is required for coordination among sensor nodes such as [33].

4.2 Protocol description

This section presents the CUMAC-CAM basic facts, which are: channel allocation matrix (CAM), propagation delay mapping, cooperative update on channel allocation, transmission scheduling along with collision detection and channel assessment. Finally, there is a discussion of projected solution of CUMAC-CAM to resolve UWSNs THT problems.

4.2.1 Channel allocation matrix (CAM) and propagation delay map database

As stated earlier, every single node keeps the information of CAM and delay map.

Channel allocation matrix (CAM) The channel allocation matrix is formulated with the aim of keeping the channel allocation information in order to maximize the network throughput. In addition, it also helps nodes to avoid collision, as nodes are aware of the occupied channel status derived from the CAM. Therefore, the main objective of maintaining the CAM is for each node to have up to date details about the network channels, which accordingly assist them in making decisions for future transmissions. Our protocol proposed the existence of multiple channels in the networks: one control channel and multiple data channels. All nodes continually listen for the control packet transmission over the control channels, and access the channel when it becomes free. As there are numerous data channels available in the networks, a crucial factor for a node to be informed prior to the upcoming transmission is channel assignments made by other neighbor nodes. The CAM holds data channel details, which consist of the employed channel information, the source information, the probable receiver intended for channel reservation, the time stamp (sent time of MAC frame) and lastly, the duration of MAC frame transmission.

For instance, the channel allocation matrix (CAM) would be identical, seen displayed in the below Table 1: occupied channels are enrolled in the CAM once the channels are reserved by the node. Then, the other essential information such as source and destination node, denoted by X1 and Y1, time stamp (T1), and the total approximate transmission time duration (D1), also tracked down on the CAM against the allotted channel. In summation, the major function of the CAM is to provide the occupied channel with information. In particular, if any node desires to transmit data, it first checks the channel allocation matrix in order to get the relevant channel reservation details. Then, after consideration of the delay mapping, it moves to the next step that is taking the initiative for further transmissions.

Delay map Each node also needs to maintain a delay map by means of passively overhearing transmissions from neighbor nodes. The delay map stores both source

information and destination information, based on the observation of MAC frames. It also contains the probable delay concerning source and destination node. Thus, the suggested CUMAC-CAM protocol confirms the time synchronization approximation among the nodes in a network similar to the DOTS protocol [14]. Thus, the aim of the delay map is to estimate the transmission delay between a sender and receiver pair, accompanied by a propagation delay assessment.

For example, one control channel and two data channels (F1, F2) are available in the proposed network model. As shown in Fig. 5, there is an ongoing transmission on the data channel F2 from the source node A to node B. In the meantime, node D and node C are handshaking on the control channel, in order to reserve the data channel F1 and continue their transmission with the desired data channel, after successful negotiation. The other remaining nodes keep listening to the control channel and update their CAM accordingly. These entries in the CAM, along with the delay map, are exploited for any future transmission.

4.2.2 Cooperative update on channel allocation

A new scheme of cooperative update mechanism on channel allotment is presented to handle the multi-channel problem of UWSN. Figure 5 illustrates that, node A and node B missed out the RTS/CTS handshaking in the middle of node D and node C because of multi-channel hidden terminal problem. As this set of nodes was in an ongoing transmission on the data channel F2, they are not informed about both handshaking and facts of upcoming transmission. Neighbor node of node A and node B has the handshaking details. Additionally, neighbor nodes also are accounted with when transmission between the nodes (A and B) will finish. Accordingly, after each ongoing transmission completes, neighbor nodes perform as helper nodes by collaborating communication sets through control packets (UPDATE packet) as depicted in Fig. 6. The objective is alleviating the UWSN problems caused by multi-channel hidden terminal by providing channel allocation update status.

The packet format of UPDATE packet is shown in Fig. 7. At the end of the prior transmission and before initiating a future transmission, the communication pairs wait up to maximum propagation delay for getting the UPDATE packets from their respective neighbor nodes. After receiving the UPDATE packets, node A and node B will update their CAM and estimate the remaining duration by considering the propagation delay between their respective neighbor nodes.

In order to mitigate UPDATE packet collision, neighbor nodes implement random back-off algorithm while

Table 1 Overview of channel allocation matrix

Allocation	Source	Destination	Time Stamp	Duration
Channel 1	X1	Y1	T1	D1
Channel 2	-	-	-	-
Channel 3	-	-	-	-

Fig. 5 Nodes are maintaining CAM and delay map database by overhearing neighbor nodes transmission

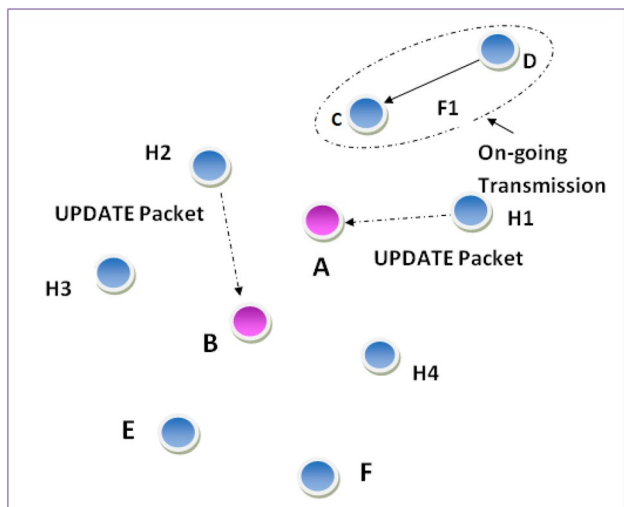
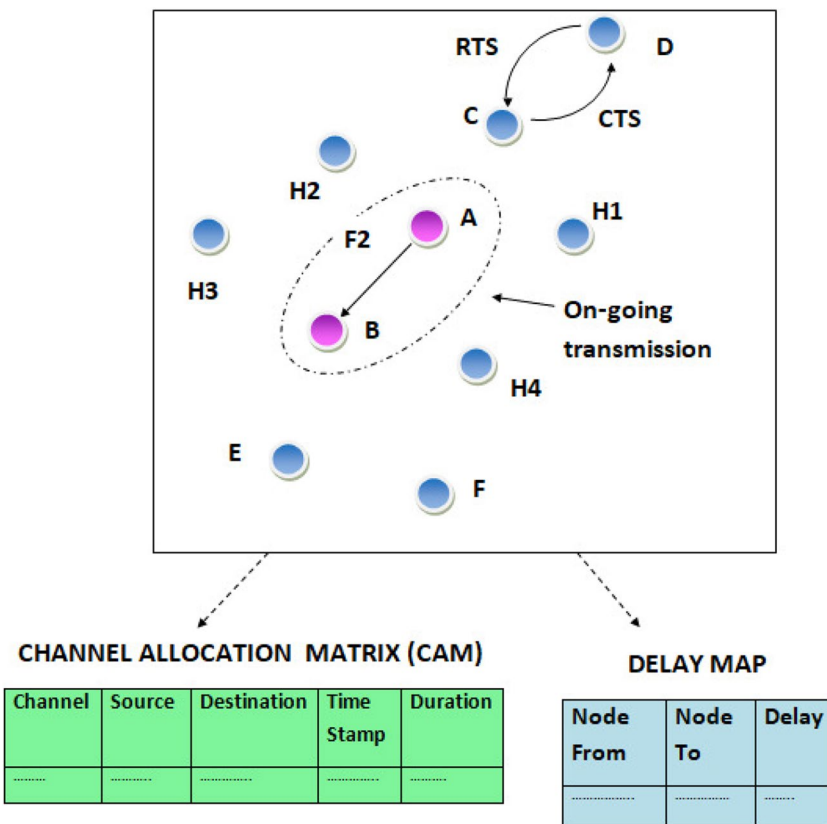


Fig. 6 UPDATE packets sent by neighbor nodes

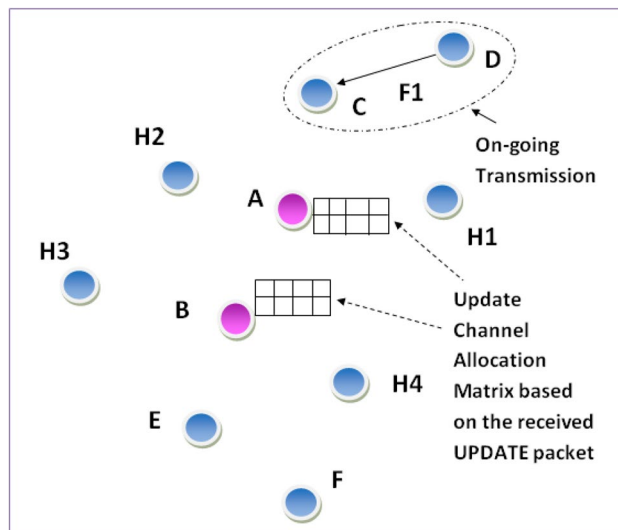


Fig. 8 CAM update based on the received UPDATE packet

Packet Type	Packet Length	Sender ID	Sender Location	Receiver ID	Receiver Location	Selected Data Channel	Data Packet Length	Remaining Duration

Fig. 7 UPDATE packet

sending UPDATE packets at the event of cooperative channel allocation update. Thus, node A and B update their respective CAM from their cooperative nodes, as represented in Fig. 8. Furthermore, work flow of cooperatively updating information of channel allocation is portrayed in Fig. 9.

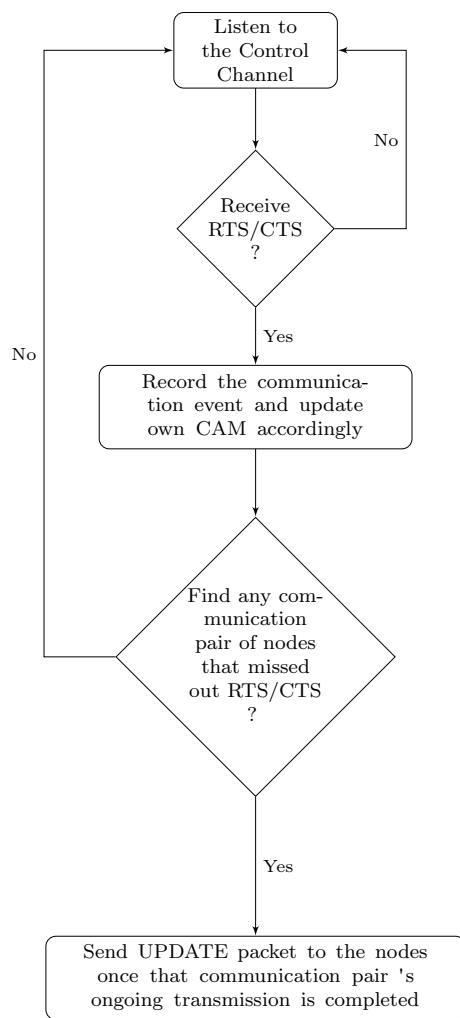


Fig. 9 The work flow of cooperative update on channel allocation

4.2.3 Transmission scheduling with collision detection and channel assessment

Whenever a node has a frame to send, at first it checks its CAM to select a data channel. If there is any free data channel is observed after checking the channel allocation in CAM, then this phase is straightforward. That is, sender node directly moves for RTS/CTS negotiation on control channel. To avoid collision on control channel that is if a sender or receiver node found that control channel is not available for control packets then they goes to backoff and tries again after a random period of time. An example of RTS/CTS collision scenario is shown in Fig. 10. It represents that, information of RTS which is sent from node b to node v, reaches to node k after experiencing a long propagation delay. At that time node k

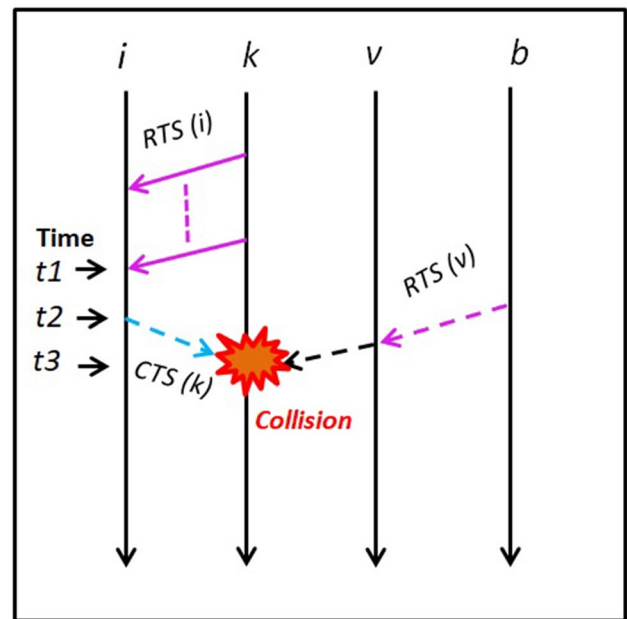


Fig. 10 RTS/CTS packet collision

was receiving the CTS from node i. As a result, control packet collision occurs at node k at time t3.

On the other hand, if data channels are found occupied then nodes have to run the transmission scheduling algorithm according to delay mapping information. The objective is to take decision for data transmission over the data channel as [14]. In the proposed scheme, sender and receiver both nodes run the transmission scheduling algorithm to initiate an upcoming data transmission. The aim is to prevent the likelihood of collisions by taking a decision whether a node can transmit/receive without interference of neighbor node's on-going transmission or not.

In the light of this research work, for any upcoming transmission a channel negotiation process is done by the set of communication pair willing for the transmission. After a successful RTS/CTS handshaking and channel negotiation, both the sender and receiver switch to the desired data channel from the control channel. Later, both nodes shift back to the control channel after successful data transmission.

Now, this segment explicates assessment mechanism of free channel. In addition, an algorithm of transmission scheduling also presented here intended for collision detection at sender and receiver end (Algorithm 1, Table 2 gives the summary of used notation in Algorithm 1).

Table 2 Summary of the used notation in Algorithm 1

Symbol	Description
V	Set of nodes
T_{trans}	Transmission time
C_{prop_delay}	Control packet propagation delay
D_{prop_delay}	Data packet propagation delay
$T_{arrival}^r$	Data arrival time at receiver node
T_{trans}^n	Data transmission time at neighbor node
$T_{arrival}^n$	Data arrival time at neighbor node
T_{trans}^s	Data transmission time at sender node

Algorithm 1 Transmission scheduling algorithm based on delay map

```

Sender end collision detection with transmission schedule
1: for  $\forall V \in$  delay map do
2:   if frame type is DATA then
3:      $T_{arrival}^n = \text{timestamp} + T_{trans} + (2 \times C_{prop\_delay}) + D_{prop\_delay}$ 
4:     if  $T_{trans}^n \in T_{arrival}^n$  then
5:       collision detected
6:     end if
7:   end if
8: end for
Receiver end collision detection with transmission schedule
9: for  $\forall V \in$  delay map do
10:  if frame type is DATA then
11:     $T_{arrival}^r = \text{timestamp} + T_{trans} + (2 \times C_{prop\_delay}) + D_{prop\_delay}$ 
12:    if  $T_{arrival}^r \in T_{trans}^n$  then
13:      collision detected
14:    end if
15:  end if
16: end for
    
```

(a) *Free channel assessment* If there is any free data channel, which is observed through verifying the CAM, then sender node will take initiative to process RTS/CTS handshaking on control channel. The purpose is to communicate with the intended receiver, as illustrated in Fig. 11. The goal is to taking initiative for near-term

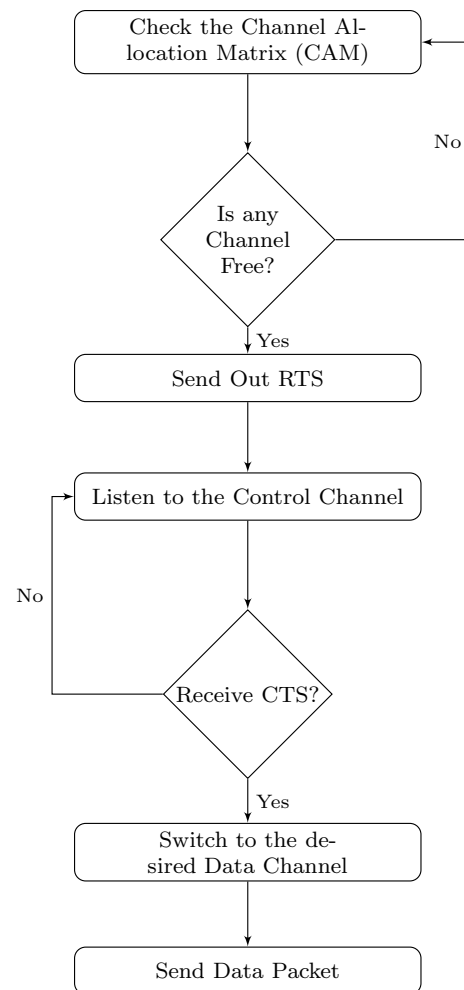


Fig. 12 The work flow of free channel assessment

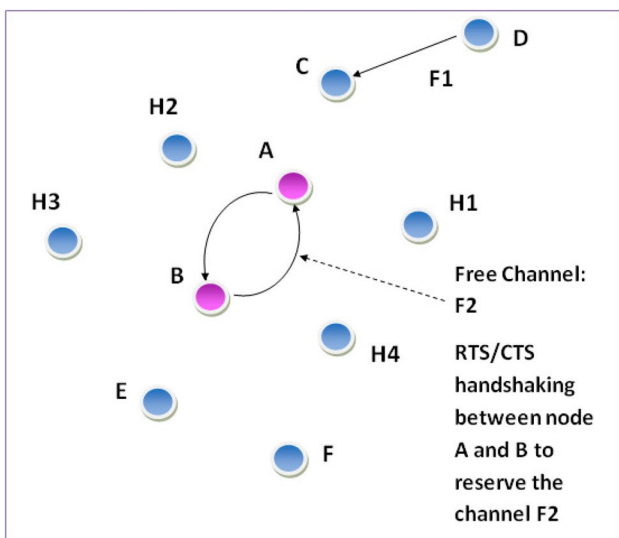


Fig. 11 RTS/CTS handshaking to reserve the channel F2 for the upcoming transmission

packet transmission on that free data channel. In this way, node itself, with the help of CAM, selects the data channel through free channel evaluation. Consequently, this eases the process of getting the data channel status. The work flow of free channel assessment is illustrated in Fig. 12.

(b) *Sender end collision detection* If all data channels are occupied that is not found free, then at first sender node checks the CAM to get the reserved channels details. After that, it executes transmission scheduling algorithm, keeping alignment with delay mapping information. Thus sender node is guided to go forward for future packet transmissions, without interruption of one hop neighbor node’s packet reception on desired channel.

Suppose, a transmission is continuing over F1 data channel and F1 is reserved by a communication set—node D (sender) and node C (receiver). Additionally, one-hop neighbor of node A is node C. In this scenario, node A has a packet to send to node B then at first, node A verifies the CAM and found that all the data channels are occupied.

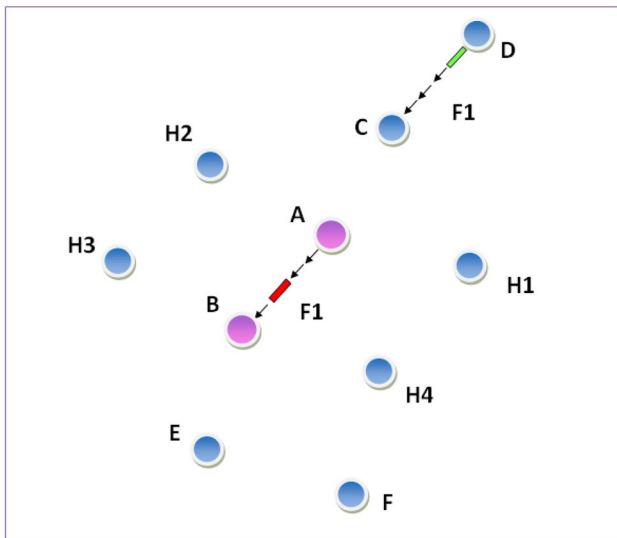


Fig. 13 Sender end collision detection. Node A transmits a packet to node B over the channel F1 before its one hop neighbor node C receives a packet from node D over the same channel

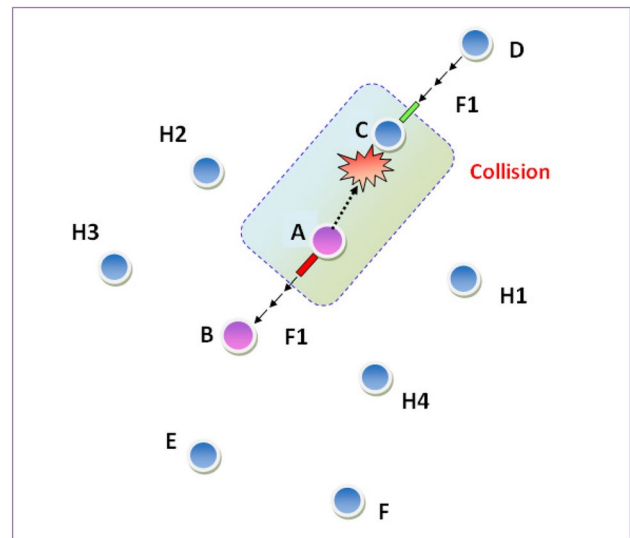


Fig. 14 Sender end collision detection. Collision occurs if node A transmit a packet to node B over the channel F1 without considering collision at one-hop neighbor node C's packet reception on the same channel

Afterwards, node A runs the transmission scheduling algorithm based on the propagation delay map to verify whether its upcoming transmission on desired data channel will create a collision with one hop neighbor nodes packet reception or not. For the given example as depicted in Fig. 13, node A will take decision for data transmission over F1 based on the information of transmission scheduling. Node A will proceed for data transmission if it gets confirmation from transmission scheduling that, node A will be able to complete transmitting data packet before node c (one hop neighbor node) receives the sent packet from node D. Else collision will take place as shown in Fig. 14. In brief, a node will go for data transmission on any reserved channel if it assess that the forthcoming data transmission will not be the reason of packet collision. To give a picture of the work flow of transmission scheduling with delay map at sender end, a pictorial representation is given in Fig. 15.

(c) *Receiver end collision detection* After receiving RTS, the receiver will go through the CAM to check the channel allocation status and executes transmission scheduling algorithm based upon delay map. Figure 16 reflects the idea, where a transmission is going on F1 data channel in the middle of communication pair—node E (sender) and node F (receiver). Node B's one-hop neighbor is node E (sender). In this context, node B could receive RTS which is sent from node A for forthcoming transmission on F1-the same channel. Then according to proposed scheme, node B will estimate whether node B's packet reception sent from node A will collide with one hop neighbor node 's (node E) packet transmission or not. Node B will send CTS

if there is a prediction of not having packet collision after running transmission scheduling algorithm. That is, node B will take delivery of incoming packet from node A after node E (one hop neighbor node of A) completes packet transmission to intend receiver-node F or else there will be collision as represented in Fig. 17. Additionally, node B will take an interval till maximum propagation delay in accordance with delay mapping information. The objective is to mitigate collision caused by long delay hidden terminal problem in UWSNs.

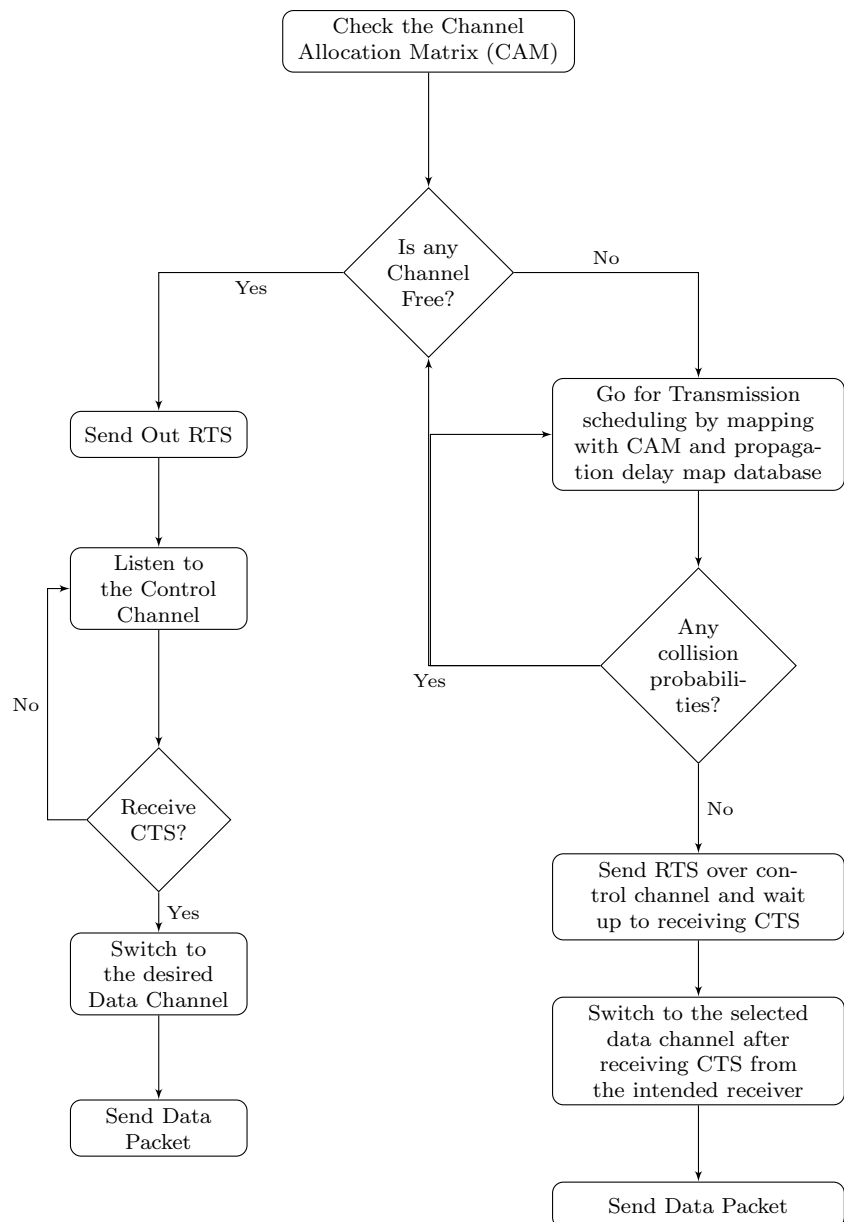
As a whole, the procedure of receiver end transmission scheduling with delay map is depicted in Fig. 18.

Thus channel assessment and transmission scheduling algorithm of the proposed scheme facilitates for collision detection as well as enhance the concurrent transmission with the benefit of delay map.

Afterward, when receiver node sends CTS to the sender, both two nodes move to the expected data channel. Then, receiver node starts a timer and waits for the incoming packet. If there is no packet to receive before time out then receiver node moves back to the control channel and keeps update its CAM. Hence, it moves forward for broadcasting a control message-CANCEL to revoke the channel reservation. In contrast, if sender node does not get any CTS after transmitting RTS packet then after random back-off period sender node retransmit the RTS packet to take initiative for transmission by way of following the proposed scheme.

(d) *Collision resolution* The above mentioned procedure reflects that there is an opportunity for simultaneous

Fig. 15 The work flow of sender end transmission scheduling with delay map



transmission in the CUMAC-CAM scheme. Nodes can estimate the channel allocation status in advance of transmission, as well as analyze the collision probabilities through the benefits of transmission scheduling algorithm, therefore, the CUMAC-CAM results in better throughput, along with the continuing transmission in parallel over the same data channel. However, nodes can experience collisions at the time of sending control packets on the control channel, if two nodes compete at the same time. It may also happen at the event of a cooperative update on channel allocation. To overcome this adverse situation, nodes that desire to send data to an intended receiver first checks the CAM to verify whether there is any free data channel or not. If any data channel is found free, the sender node sends

the RTS to the desired receiver over the control channel, in order to have the RTS/CTS handshaking. If two or more nodes contest simultaneously on control channel, there is a chance of collision. To reduce collision, nodes pick up a random back-off and try again after a random period of time. On the contrary, if the data channels are occupied, then nodes run the proposed transmission scheduling algorithm (as stated in Algorithm 1) to forecast whether its upcoming transmission will create a collision or not. Both sender and receiver nodes follow the same scheme for collision resolution at the sender and the receiver end. As each node keeps the CAM and the delay mapping information, in addition to running the transmission scheduling algorithm, this scheme provides a direction for nodes

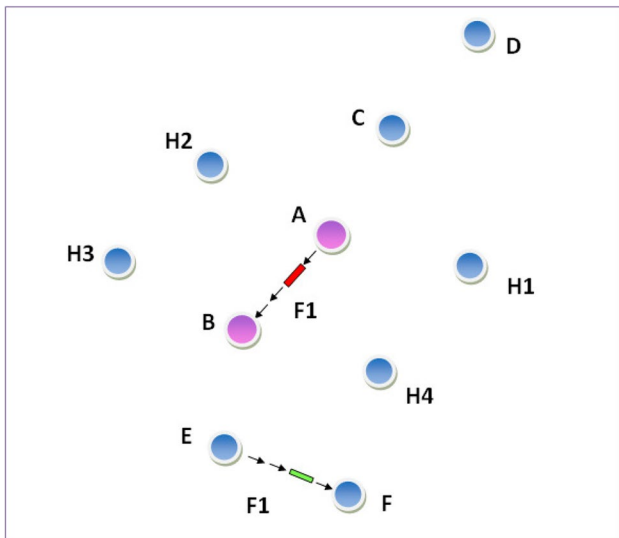


Fig. 16 Receiver end collision detection. Node B will receive packet from node A on F1 channel after node B's one hop neighbor node E transmits a packet to its intended receiver—node F over the same channel

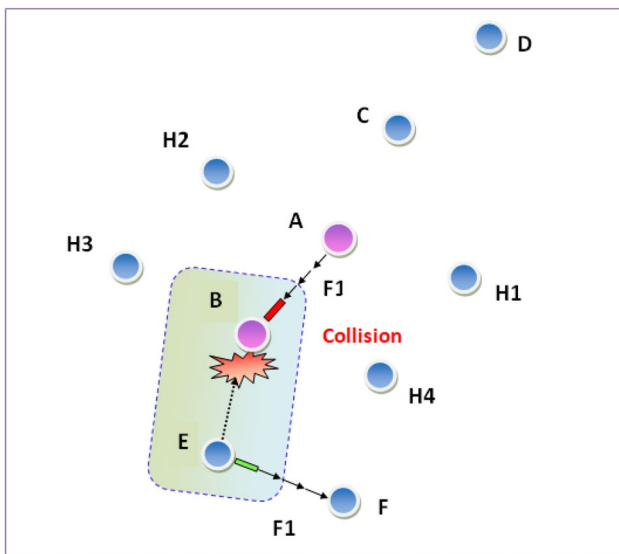


Fig. 17 Receiver end collision detection. Collision occurs if node B receives a packet over the channel F1 while its one hop neighbor node transmitting a packet to node F over the same channel

to proceed for further simultaneous transmission, without interrupting their one hop neighbor node's transmission. In short, this scheme both improves the overall network performance and moderates likelihoods of collisions. However, collisions may still occur if CAM and delay mapping information are not up to date. Therefore, as a preventive measure, the neighbor node, which is a potential helper node of an ongoing transmission, broadcasts UPDATE

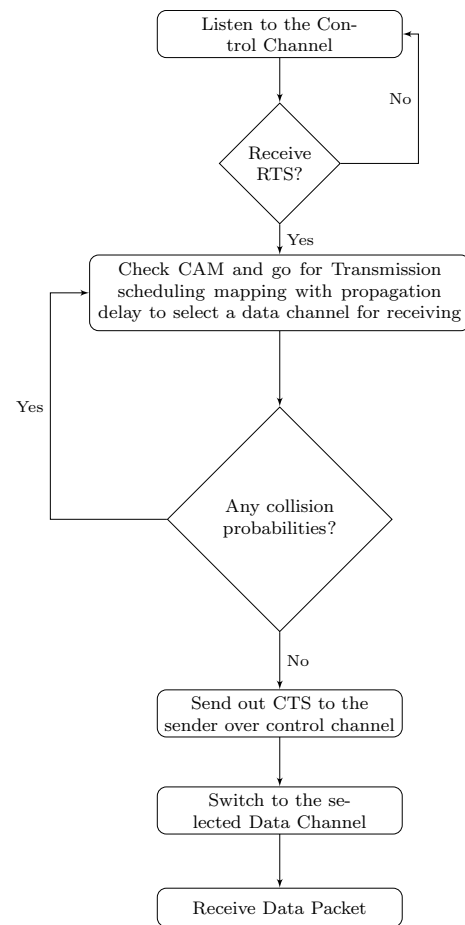


Fig. 18 The work flow of receiver end transmission scheduling with delay map

control packets to the communication pair whenever they complete their ongoing transmission. As a result, it helps to reduce the number of collisions in a network. However, UPDATE packets may also collide in the case of more than one helper node. To address this packet collision, we assume that helper nodes send UPDATE packets after taking a random back-off.

4.2.4 Discussion

The enhancement on CUMAC with the introduction of CAM, channel assessment, cooperative update on channel allocation and transmission scheduling with delay map effectively mitigate the THT problems in underwater sensor networks. In reference of CUMAC as well as research work in literature, the conventional problem of UWSN due to multi-hop hidden terminal can be eased through RTS/CTS handshaking. Hence, the key point is to handle multi-channel and long delay hidden terminal problems of UWSNs. In the proposed scheme, CAM has been presented. Every single node keeps both CAM and

a delay map by way of passively overhearing the continuing neighbor nodes transmission. A cooperative apprise on channel allocation efficiently addresses the problem related to multi-channel hidden terminal in underwater networks. As this scheme aids the communication set nodes (which was busy on their ongoing transmission on data channel and missed out the recent channel negotiation through RTS/CTS handshaking on control channel) to get details of update channel allocation information.

Furthermore, all nodes assess the channel and execute the transmission scheduling algorithm keeping map with delay mapping information. The purpose is to have successful transmission without collision at sender and receiver both end. Moreover, receiver node waits until maximum propagation delay before sending CTS to its corresponding sender to keep on data transmission over selected data channel. Therefore, the receiver node may get opportunity to have the channel detail status before the waiting time out from long delayed neighbor nodes for which the same data channel is already reserved. In this way, proposed CUMAC-CAM protocol efficiently alleviates the THT problems in UWSNs.

5 Results and performance evaluation

With the purpose of analyzing performance using simulator, the proposed protocol implemented in Aqua-Sim, one of the most popular network simulators for underwater sensor networks. Aqua-Sim is an extension of NS2 simulator for simulating underwater sensor network protocols [34].

5.1 Simulation settings

The performance of CUMAC-CAM protocol analyzed through simulation and compared with two multi-channel MAC protocols: CUMAC and RTS/CTS based multi-channel MAC protocol. The performance is evaluated in terms of the metrics as listed below:

(i) Average network throughput, (ii) average energy consumption, (iii) end to end delay, (iv) packet delivery ratio (v) packet loss ratio, (vi) collision probability and (vii) fairness index

In this simulation set, a random network is examined. The proposed CUMAC-CAM protocol is carried out in a random network where a maximum 20 static sensor nodes are uniformly disseminated in a 500m X 500m area. We also implement two other multi-channel MAC protocols: the CUMAC [18] and the RTS/CTS scheme [15] in the same network topology. There are a maximum of 8 channels in the network, but performance has been analyzed by varying channels from 2 to 8. The transmission range of every node

Table 3 System parameters

Parameters	Value
Transmission range of every node	100 m
Maximum number of channels	8
Acoustic propagation speed	1500 m/s
Simulation time	300 s
Maximum data packet length	600 bytes
Control packet length	32 bytes
Data rate	1 kbps
Transmitting power	0.6 W
Receiving power	0.2 W
Number of simulation run	100

is 100 m, the acoustic propagation speed 1500 m/s, and the data rate is 1 kbps. The maximum data packet length is 600 bytes, but it varied from 200 to 600 bytes for performance evaluation. The control packet length is 2 bytes. The average transmitting and receiving power are set to 0.6 W and 0.2 W respectively. Next, the propagation delay is varied from 0.1 to 0.7 s for the purpose of comparison. Moreover, we vary both the traffic load, as well as the number of nodes, for this performance assessment. Unless otherwise specified, the number of hops for this simulation is set to 2. Input traffic is kept at 0.02 packets per second for all the results. Each simulation executes for 300 s, and a series of 100 simulation runs have been carried out for the purpose of performance evaluation. However, most of the simulation parameters are taken from [15, 18]. In Table 3, key simulation parameters are presented.

5.2 Simulation results

5.2.1 Average network throughput

Average network throughput denoted as average number of successful transmitted data bytes per second

Average Network Throughput

$$= \frac{\text{Average of total transmitted data}}{\text{Network operation time}} \quad (1)$$

The performance of CUMAC, RTS/CTS scheme and CUMAC-CAM protocol has been evaluated in terms of input traffic, number of channels, data packet length, delay variance and number of hops and the details comparison is given in the following section.

(a) *Impact of input traffic* Figure 19 depicts that, throughput increases with input traffic in case of CUMAC-CAM and for the other two protocols—CUMAC and RTS/CTS scheme. With this simulation set it is observed that, CUMAC-CAM can attain higher throughput in comparison with CUMAC

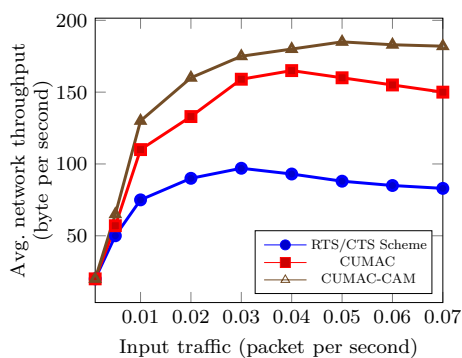


Fig. 19 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of input traffic on average network throughput

and RTS/CTS scheme. The throughput of CUMAC-CAM protocol is around 175 bytes per second where CUMAC can reach its highest throughput nearby 165 bytes per second at input traffic is 0.04 packets per second. On the other hand, CUMAC-CAM achieves around 170 bytes per second when RTS/CTS scheme attain its maximum throughput 95 bytes per second at 0.03 packets per second input traffic. From the graph it is observed that, throughput gradually drops along with the input traffic for all three protocols. As collision probability increases along with input traffic therefore it impacts the throughput to decrease accordingly. In case of CUMAC-CAM as Fig. 19 represents, throughput goes to the utmost peak 180 bytes per second at 0.05 packets per second. Subsequently, its throughput drops gradually but delivers relatively firm performance compared to other two protocols. The reason behind is, CUMAC-CAM implements channel allocation assessment with channel allocation matrix and transmission scheduling algorithm to avoid the collision. As probability of network collision decreases, CUMAC-CAM results better performance than CUMAC and RTS/CTS scheme.

(b) *Impact of number of channels* In this simulation setting, performance of all three protocols- proposed CUMAC-CAM, CUMAC and RTS/CTS scheme has been evaluated through varying number of data channels. Performance has been assessed by varying channels starting from 2 to 8. As per Fig. 20, network throughput upturns with number of channels in all three cases. The reason is the implementation of several data channels in the network. If there are multiple data channels in network then the net input traffic to each data channel will be decreased. Hence, there will be less collision probability and higher throughput. Furthermore, Fig. 20 represents that, in comparison of CUMAC and RTS/CTS protocol CUMAC-CAM is more effective. This is because, channel allocation assessment, channel employment, collision detection with transmission scheduling algorithm scheme of the CUMAC-CAM

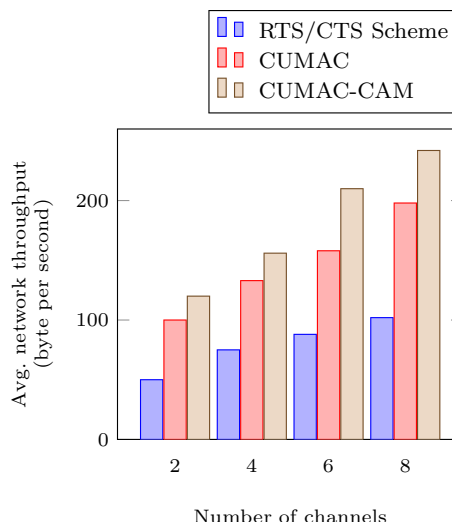


Fig. 20 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of number of channels on average network throughput

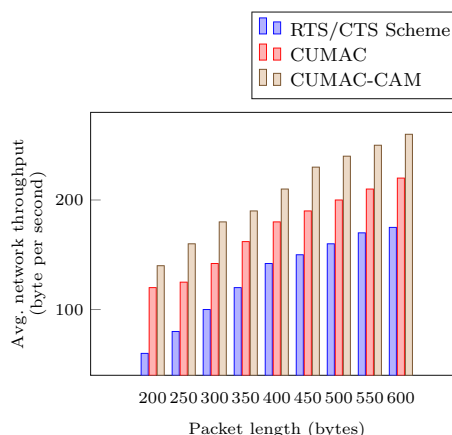


Fig. 21 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of packet length on average network throughput

protocol results less collision probability, efficient channel allotment, as a final provides higher throughput.

(c) *Impact of data packet length* In order to compare the performance of three protocols with regard to varying packet length, data packet length is changed from 200 to 600 bytes. Figure 21 shows that, network throughput improves with the packet length for CUMAC-CAM and the other two protocols. In case of longer data packet, one data packet transmission contributes more throughputs. In case of longer data packet, one data packet transmission contributes more throughputs. Thus, network throughput increases with longer data packet length. In this simulation settings, CUMAC-CAM also attain higher throughput than CUMAC and RTS/CTS scheme. CUMAC-CAM's throughput

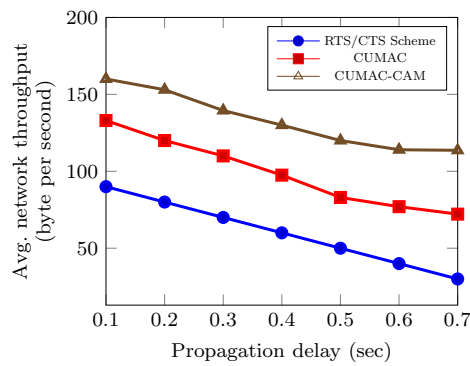


Fig. 22 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of propagation delay on average network throughput

reaches to 260 bytes per second when the maximum data packet length is 600 bytes. However, the throughput of CUMAC and RTS/CTS scheme stands for 220 and 175 bytes per second respectively with this data packet size. The reason behind of this significance is, less collision probability in CUMAC-CAM protocol. As longer data packet may experience high collision possibility and CUMAC-CAM proposes less collision than the other two. Therefore, CUMAC-CAM offers moderately significant performance than CUMAC and RTS/CTS scheme.

(d) *Impact of delay variance* For the performance comparison in terms of delay variance, propagation delay varies from 0.1 to 0.7 s. In this set of simulations, the average distance of nodes varies from 50 to 100 m. Propagation delay can be calculated from the distance between nodes by dividing the propagation velocity of acoustic wave (1500 m/s). Figure 22 plots that, network throughput decreases noticeably with the increase of propagation delay. Furthermore, it is observed that, CUMAC-CAM outperforms the other two protocols with delay variance. CUMAC-CAM protocol implements delay map and channel allocation matrix as well as run transmission scheduling to ensure collision free parallel transmission. On the other hand, CUMAC does not consider the propagation delay for continuing parallel transmission and RTS/CTS scheme does not focus on long delay hidden terminal problem. Therefore, CUMAC-CAM provides better performance, with varying propagation delay, compared to CUMAC and RTS/CTS scheme.

5.3 Average energy consumption

Average energy consumption is defined by dividing network's total energy consumption by successful transmitted data bytes and it is measured by milli-joule per byte.

$$E_{avg} = \frac{E_{consumption}}{\text{Total transmitted data}} \quad (2)$$

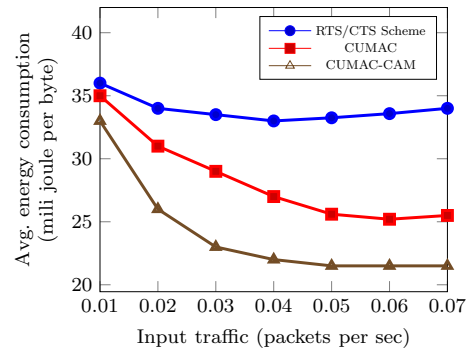


Fig. 23 Comparison of the CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of input traffic on average energy consumption

E_{avg} refers to average energy consumption per byte ; $E_{consumption}$ refers to total energy consumption in the network.

(a) *Impact of input traffic* Figure 23 provides an analysis of average energy consumption among three protocols. It is clear that, average energy consumption per byte drops along with input traffic upturn for CUMAC-CAM and CUMAC protocol. But it is noticed that, energy consumption increases with the input traffic for RTS/CTS scheme. In case of CUMAC and CUMAC-CAM protocol, less energy will be expended for idle listening with the increase of input traffic and energy consumption decreases accordingly. When comparing three protocols, it is observed that there has been a significant energy efficiency of proposed CUMAC-CAM protocol than the other two. The reason behind is, cooperative collision detection and tone pulse sequence mechanism of CUMAC subdue the UWSNs triple hidden terminal problems. Oppositely, RTS/CTS scheme does not consider the hidden terminal problems which occurred for multichannel and long propagation delay. As a result, collision increases with the increase of input traffic and it impacts energy consumption. In summary, CAM, propagation delay mapping and transmission scheduling along with collision detection appliance of CUMAC-CAM efficiently defeats the triple hidden terminal problems in UWSNs. Thus, it drives providing better output than the other two protocols- CUMAC and RTS/CTS scheme.

(b) *Impact of number of channels* Impact of data channels has also been taken under consideration to analyze the energy efficiency improvement with the number of channels. As more the data channels, less the collision probability on the data channels.

In this set of simulations, performance has been analyzed through varying number of data channels from 2 to 8. Figure 24 illustrates that, CUMAC-CAM acquires higher energy efficiency in comparison of CUMAC and RTS/CTS scheme. Energy efficiency also improves according to

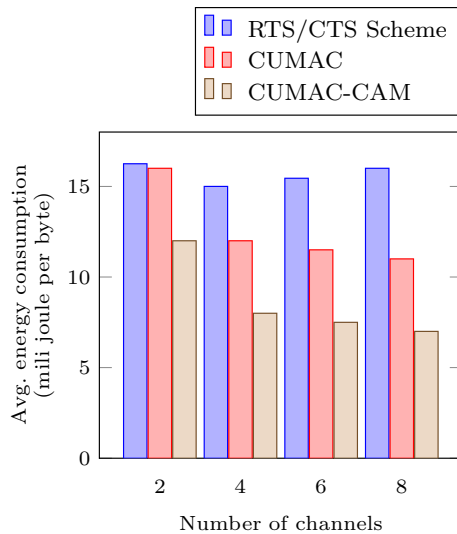


Fig. 24 Comparison of the CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of number of channels on average energy consumption

number of channels. It can be seen from the bar chart that, there is a fluctuant trend in average energy consumption for RTS/CTS scheme, while the other two protocols-CUMAC-CAM and CUMAC demonstrated in downward trend. In case of data channel=2, average energy consumption started at 16 milli-joules per byte for RTS/CTS scheme, 15.75 milli-joules per byte for CUMAC, 12 milli-joules per byte for CUMAC-CAM. On the other hand, when there are 8 data channels, average energy consumption decreases to 15 milli-joules per byte for RTS/CTS scheme, 11 and 7 milli-joules per byte for CUMAC and CUMAC-CAM respectively. Therefore, it is observed that, CUMAC-CAM performs with significant energy efficiency. The reason behind is, multiple data channels association reduces the likelihood of collision which consequently reflects in network performance. However, average energy consumption is much higher for RTS/CTS scheme than CUMAC-CAM and CUMAC, as there is higher collision probability for multi-channel hidden terminal problems.

(c) *Impact of data packet length* To analyze the performance results concerning variable packet length, different length of data packet has been implemented started from 200 to 600 bytes. The performance results are illustrated in Fig. 25. From the bar chart of Fig. 25, it is noticeable that, energy consumption decreases with regard to upward packet length trend for CUMAC-CAM and CUMAC protocols. In case of RTS/CTS scheme, average energy consumption monotonically moves to downward with increasing data packet length. From performance results, it is visible that CUMAC-CAM conquers remarkable energy efficiency compared to CUMAC and RTS/CTS scheme. Average energy consumption reduces to 5

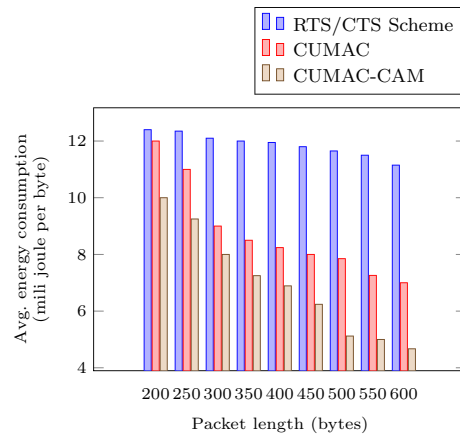


Fig. 25 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of packet length on average energy consumption

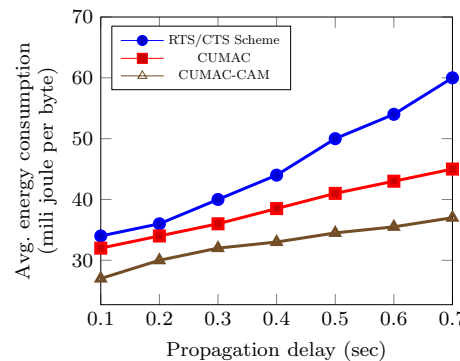


Fig. 26 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of propagation delay on average energy consumption

and 7 milli-joules per byte for CUMAC-CAM and CUMAC protocol respectively. On the contrary, in case of RTS/CTS scheme, average energy consumption reduces to 11 milli-joules per byte when the data packet length changes to maximum length 600 bytes. The reason behind of this significance is, less collision probability in CUMAC-CAM. Therefore, the chances of collision probability increase in parallel of increasing data packet length and CUMAC-CAM offers less collision than the other two. In summary, CUMAC-CAM delivers better output in comparison of CUMAC and RTS/CTS scheme.

(d) *Impact of delay variance* Figure 26 shows the performance comparison of average energy consumption in terms of delay variance, propagation delay varies from 0.1 to 0.7 s. In this set of simulations, the average distance of nodes varies from 50 to 100 m. As seen from the graph, average energy consumption increases with the increase of propagation delay. In addition, it is observed that, CUMAC-CAM outperforms the other two protocols

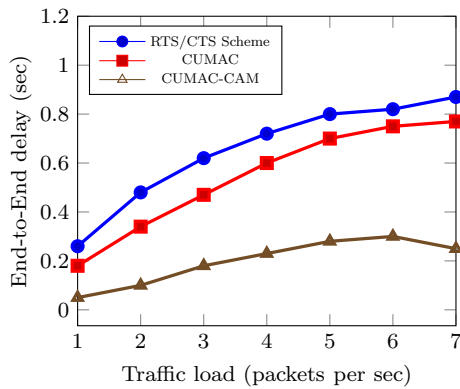


Fig. 27 Comparison of the CUMAC-CAM protocol with CUMAC and RTS/CTS scheme in terms of impact of traffic load on end to end delay

with delay variance. As CUMAC-CAM exploits delay mapping prior to originating a transmission therefore it facilitates a node to forecast the collision possibility of its upcoming packet transmission. However, CUMAC does not consider the propagation delay for continuing concurrent transmission and RTS/CTS scheme does not focus on long delay hidden terminal problem. Thus, CUMAC-CAM still provides better performance, with varying propagation delay, compared to CUMAC and RTS/CTS scheme.

5.4 End to end delay

End-to-end delay can be defined as—each packet’s total required time to reach from source to destination on average. It includes all types of delays experienced through trip starting from sender to receiver.

$$End_to_End_Delay = \frac{\sum_{n=1}^N (R_n - S_n)}{N} \tag{3}$$

Here, S_n Sending time of n th packet, R_n receiving time of n th packet, N number of packets received.

Impact of traffic load Figure 27 represents the end-to-end delay performance among CUMAC, RTS/CTS scheme and CUMAC-CAM. From this graphical representation it is observed that, CUMAC-CAM attains lowest end-to-end delay than CUMAC and RTS/CTS scheme. As presented in the line graph, for all three protocols delay gradually moves upwards. This is because, initially node gets available free data channel for data transmission. After that, data channels are started to reserve and later those release again. Hence, the line graph shows that, end-to-end delay progressively increases along with high data channel access contention and dense traffic load. Performance results show that, CUMAC-CAM provides much better

Table 4 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of packet delivery ratio

No. of NNodes	CUMAC PDR %	RTS/CTS scheme PDR %	CUMAC-CAM PDR %
5	87.5714	68.4019	91.7143
10	87.5620	67.0012	91.0014
20	86.0012	65.5578	90.0004

performance than the other two. As CUMAC-CAM allows concurrent transmission with the mechanism of channel allocation assessment, propagation delay mapping and transmission scheduling with collision detection thus eases the queuing delay through concurrent transmission and accomplishes better output for delay performance.

5.5 Packet delivery ratio (PDR)

The packet delivery ratio is the ratio of successfully carried through packets at the destination to the packets generated by the source. In short, it is the success rate of the protocol from source to destination. It can be denoted as:

$$PDR = \frac{\text{Number of received packets}}{\text{Number of generated packets}} \times 100 \tag{4}$$

Table 4 represents that, PDR decreases when number of nodes is increased. In short, CUMAC-CAM provides better result than CUMAC and RTS/CTS scheme protocol with reference to packet delivery ratio.

5.6 Packet loss ratio (PLR)

Packet loss happens when one or more packets fail to reach their destination during data transmission across the network. Packet loss can be measured by a percentage of packets lost with respect to packets sent. Therefore,

$$PLR = \frac{\text{Number of lost packets}}{\text{Number of sent packets}} \times 100 \tag{5}$$

As Table 5 represents, packet loss ratio increases with proportion to network size. With the network size, input traffic as well as collision probability in the network increases which causes packet loss. It is observed from the packet loss ratio analysis that, CUMAC-CAM gives better performance than CUMAC and RTS/CTS scheme. This is because, channel allocation using propagation delay map along with transmission scheduling mechanism facilitates CUMAC-CAM to manage input traffic load, channel utilization and collision detection. As a result, performance of CUMAC-CAM provides better output.

Table 5 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of packet loss ratio

No. of nodes	CUMAC PLR %	RTS/CTS scheme PLR %	CUMAC-CAM PLR %
5	10.4286	28.5981	7.2857
10	10.4380	29.1181	8.0086
20	11.9999	30.0012	8.1281

5.7 Collision probability

The packets (control/data packets) that are transmitted within a time period are not successfully received by the intended receiver due to collision. As it is mentioned earlier, there is one control channel and multiple data channels in CUMAC, RTS/CTS scheme and the proposed CUMAC-CAM. In case of CUMAC-CAM, if a node detects that control channel is not available for packets then it goes to backoff and tries again after a random period of time. But collision might happen for packets due to long propagation delay.

Although CUMAC-CAM offers the mechanism of channel allocation assessment, transmission scheduling mapping with CAM and propagation delay to alleviate collision and support parallel transmission but collision might happen if multiple data packets are transmitted or received in the same data channel for a particular period of time. The collision probability can be calculated as:

$$P_{Col} = 1 - S_{Success Rate} \tag{6}$$

Here, P_{Col} is the packet collision probability; $S_{Success Rate}$ is obtained by dividing the number of packets successfully received by number of generated packets in the total simulation time. Figures 28 and 29 show the control packet and data packet collision probability curves with respect to arrival rate and this comparison reflects that CUMAC-CAM provides better output than the other two protocols. As CUMAC-CAM goes for backoff algorithm for control packets transmission and implements channel allocation assessment, transmission scheduling mapping with CAM and delay mapping for data packet transmission. The aim is to avoid collisions. Therefore, results reflect that CUMAC-CAM protocol experiences less collision probability compared to CUMAC and RTS/CTS scheme.

5.8 Fairness index

In underwater sensor networks, the propagation of acoustic signal is about 1500 m/s, leading the propagation delay of UWSN being several times longer than terrestrial wireless network. Furthermore, underwater sensor networks

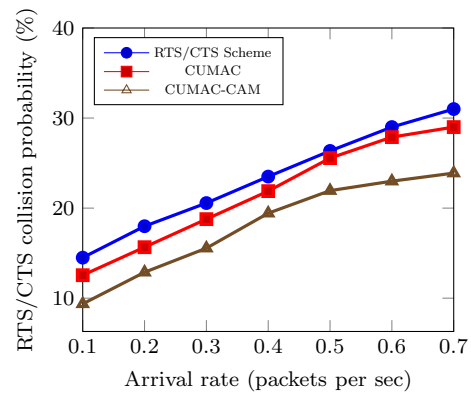


Fig. 28 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of arrival rate on control packet collision probability

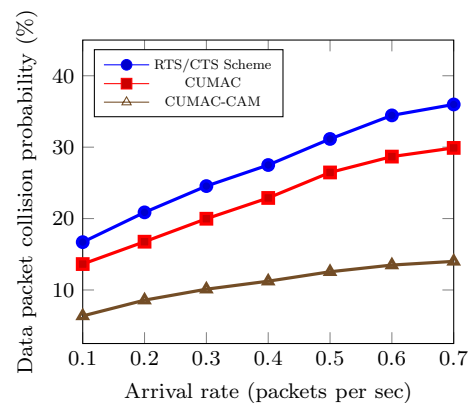


Fig. 29 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of impact of arrival rate on data packet collision probability

face space-time uncertainty problem due to this high propagation delay [35, 36]. Consequently, spatial fairness becomes a challenging problem in UWSNs. The packet which is sent earlier may reach later to the intended receiver due to the location and propagation delay. Closer nodes to the receiver have the opportunities to occupy the channel quickly, leads unfairness among the nodes. To analyze the fairness, Jain’s fairness index [14] is used and defined as:

$$Fairness\ index = \frac{(\sum x_i)^2}{n \times (\sum x_i^2)} \tag{7}$$

where x_i and n represents the throughput of node i and the number of nodes in the network respectively.

Figure 30 represents the comparison of fairness index among the three protocols. It is noticeable that, the CUMAC-CAM protocol shows (0.9 and above) high

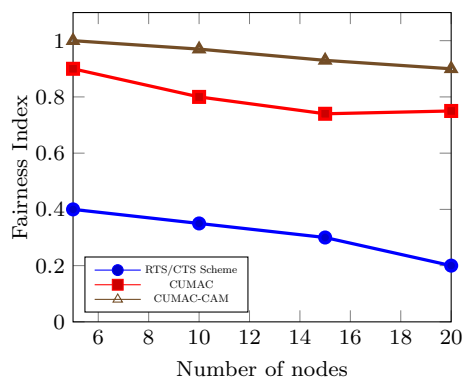


Fig. 30 Comparison of CUMAC-CAM with CUMAC and RTS/CTS scheme in terms of fairness index

fairness index and stay stable with the network size. As the proposed CUMAC-CAM protocol use channel allocation matrix with propagation delay and timestamp information during transmission scheduling. Therefore, every sender-receiver has a fair opportunity of concurrently accessing the medium with the proposed scheme. On the other hand, CUMAC implements cooperative collision detection and tone pulse sequence. But CUMAC and RTS/CTS scheme both protocols do not consider the timestamp and propagation delay mapping during transmission. Consequently, CUMAC-CAM achieves higher fairness compared to CUMAC and RTS/CTS scheme.

As a whole, simulation results verify that CUMAC-CAM significantly improves the network performance and reliability of the system in terms of throughput, energy consumption, end-to-end delay, packet delivery ratio, packet loss ratio, collision probability and fairness index compared to CUMAC and RTS/CTS scheme protocols.

6 Conclusions

CUMAC-CAM offers an effective solution to mitigate triple hidden terminal problems in UWSNs through an efficient channel selection and collision detection scheme. With the benefit of propagation delay, CUMAC-CAM put emphasis on increasing the concurrent transmission probability as well as preventing the chances of collisions. Analyzing the performance results, it is evident that, CUMAC-CAM is more efficient and provides better results with respect to energy consumption, network throughput, end to end delay, packet delivery ratio, packet loss ratio, collision probability and fairness index compared to the contemporary CUMAC and RTS/CTS scheme protocols.

But there are some limitations too, such as each node maintains a CAM and delay map to assess channel to make a decision for collision free transmission. Therefore, an

additional memory allocation is needed for the maintenance of CAM and delay map. Moreover, channel allocation assessment strategy is not implemented for control packet transmission, which may lead to collisions over the control channel. Thus to provide an effective solution for avoiding control packets collision, an investigation is required to employ transmission scheduling with delay mapping strategy on control channel as implemented for data channels in proposed CUMAC-CAM protocol.

Acknowledgements The authors would like to thank Shamim Ara Shawkat of University of Tennessee for her helpful advice on preparing the preliminary version of this paper.

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interest.

References

- Gkikopouli A, Nikolakopoulos G, Manesis S (2012) A survey on underwater wireless sensor networks and applications. In: 2012 20th mediterranean conference on control & automation (MED). IEEE, pp 1147–1154
- Casari P, Zorzi M (2011) Protocol design issues in underwater acoustic networks. *Comput Commun* 34(17):2013–2025
- Yunus F, Ariffin SHS, Zahedi Y (2010) A survey of existing medium access control (MAC) for underwater wireless sensor network (UWSN). In: 2010 Fourth Asia international conference on mathematical/analytical modelling and computer simulation (AMS). IEEE, pp 544–549
- Bouabdallah F, Zidi C, Boutaba R, Mehaoua A (2018) Collision avoidance energy efficient multi-channel MAC protocol for underwater acoustic sensor networks. *IEEE Trans Mob Comput*. <https://doi.org/10.1109/TMC.2018.2871686>
- Rahman P, Alam MS, Shawkat SA, Hoque MA (2017) CUMAC-CAM: addressing triple hidden terminal problems for multi-channel transmission in underwater sensor networks. In: Proceedings of the international conference on underwater networks & systems. ACM, p 1. <https://doi.org/10.1145/3148675.3148707>
- Akyildiz IF, Pompili D, Melodia T (2004) Challenges for efficient communication in underwater acoustic sensor networks. *ACM Sigbed Rev* 1(2):3–8
- Akyildiz IF, Pompili D, Melodia T (2005) Underwater acoustic sensor networks: research challenges. *Ad Hoc Netw* 3(3):257–279
- Burrows GE, Khan JY (2010) Investigation of a short-range underwater acoustic communication channel for MAC protocol design. In: 2010 4th international conference on signal processing and communication systems (ICSPCS). IEEE, pp 1–8
- Khan G, Dwivedi RK, Gola KK (2019) Comparative analysis of routing algorithms for underwater sensor network. In: Bhargava D, Vyas S (eds) *Pervasive computing: a networking perspective and future directions*. Springer, Berlin, pp 31–46
- Partan J, Kurose J, Levine BN (2007) A survey of practical issues in underwater networks. *ACM SIGMOBILE Mob Comput Commun Rev* 11(4):23–33
- Syed AA, Ye W, Heidemann J (2008) T-Lohi: a new class of MAC protocols for underwater acoustic sensor networks. In: IEEE INFOCOM 2008. The 27th conference on computer communications. IEEE, pp 231–235

12. Hsu C-C, Liu H-H, Gomez JLG, Chou C-F (2015) Delay-sensitive opportunistic routing for underwater sensor networks. *IEEE Commun Sens J* 15(11):6584–6591
13. Li C, Xu Y, Xu C, An Z, Diao B, Li X (2015) Dtmac: a delay tolerant MAC protocol for underwater wireless sensor network. *IEEE Sens J* 16:4137–4146
14. Noh Y, Lee U, Han S, Wang P, Torres D, Kim J, Gerla M (2014) DOTS: a propagation delay-aware opportunistic MAC protocol for mobile underwater networks. *IEEE Trans Mob Comput* 13(4):766–782
15. Zhou Z, Peng Z, Cui J-H, Shi Z (2008) Analyzing multi-channel MAC protocols for underwater acoustic sensor networks. The Computer Science & Engineering, University of Connecticut, Storrs, CT, USA, UCONN CSE technical report of UbiNet-TR08-02
16. Han YS, Deng J, Haas ZJ (2006) Analyzing multi-channel medium access control schemes with ALOHA reservation. *IEEE Trans Wirel Commun* 5(8):2143–2152
17. Yu Y, Shi J, He K (2014) A multichannel MAC protocol for underwater acoustic sensor networks. In: WUWNET'14. International conference on underwater networks & systems. ACM, pp 1–5
18. Zhou Z, Peng Z, Cui J-H, Jiang Z (2012) Handling triple hidden terminal problems for multichannel MAC in long-delay underwater sensor networks. *IEEE Trans Mob Comput* 11(1):139–154
19. Yishan S, Jin Z (2016) UMMAC: a multi-channel MAC protocol for underwater acoustic networks. *J Commun Netw* 18(1):75–83
20. Chen K, Ma M, Cheng E, Yuan F, Wei S (2014) A survey on MAC protocols for underwater wireless sensor networks. *IEEE Commun Surv Tutor* 16(3):1433–1447
21. Ansari S, Gonzalez JP, Otero P, Ansari A (2015) Analysis of MAC strategies for underwater applications. *Wirel Pers Commun* 85(2):359–376
22. Liao W-H, Lin Y-C, Kuai S-C (2014) A receiver-initiated MAC protocol for underwater acoustic sensor networks. In: The international conference on information networking 2014 (ICOIN2014). IEEE, pp 1–6
23. Peng Z, Zhu Y, Zhou Z, Guo Z, Cui J-H (2010) COPE-MAC: a contention-based medium access control protocol with parallel reservation for underwater acoustic networks. In: OCEANS 2010 IEEE-Sydney. IEEE, pp 1–10
24. Kredon II K, Djukic P, Mohapatra P (2009) STUMP: exploiting position diversity in the staggered TDMA underwater MAC protocol. In: INFOCOM 2009, IEEE. IEEE, pp 2961–2965
25. Moon E, Khan MTR, Park H, Ahmed SH, Lee S, Kim D (2018) OMRI-MAC: optimized multi-transmission receiver-initiated MAC in underwater wireless sensor networks. *Wirel Pers Commun*. <https://doi.org/10.1007/s11277-018-5977-x>
26. Jiang SM (2018) State-of-the-art medium access control (MAC) protocols for underwater acoustic networks: a survey based on a mac reference model. *IEEE Commun Surv Tutor* 20(1):1
27. Feng X, Wang Z, Han G, Qu W, Chen A (2018) Distributed receiver-oriented adaptive multichannel MAC for underwater sensor networks. *IEEE Access* 6:11666–11675
28. Tan LT, Le LB (2014) Joint cooperative spectrum sensing and MAC protocol design for multi-channel cognitive radio networks. *EURASIP J Wirel Commun Netw* 2014(1):101
29. Tan LT, Le LB (2012a) Channel assignment with access contention resolution for cognitive radio networks. *IEEE Trans Veh Technol* 61(6):2808–2823
30. Tan LT, Le LB (2012b) Fair channel allocation and access design for cognitive ad hoc networks. In 2012 IEEE global communications conference (GLOBECOM). IEEE, pp 1162–1167
31. Chao C-M, Wang Y-Z, Ming-Wei L (2012) Multiple-rendezvous multichannel MAC protocol design for underwater sensor networks. *IEEE Trans Syst Man Cybern Syst* 43(1):128–138
32. Alfouzan FA, Shahrabi A, Ghoreyshi SM, Boutaleb T (2019) A collision-free graph coloring MAC protocol for underwater sensor networks. *IEEE Access* 7:39862–39878
33. Han G, Jiang J, Shu L, Yongjun X, Wang F (2012) Localization algorithms of underwater wireless sensor networks: a survey. *Sensors* 12(2):2026–2061
34. UWSN Research Group (2012) User tutorial for aqua-sim. University of Connecticut, Storrs
35. Liao W-H, Huang C-C (2012) SF-MAC: a spatially fair MAC protocol for underwater acoustic sensor networks. *IEEE Sens J* 12(6):1686–1694
36. Hsu C-C, Lai K-F, Chou C-F, Lin KC-J (2009) ST-MAC: spatial-temporal MAC scheduling for underwater sensor networks. In: INFOCOM 2009, IEEE. IEEE, pp 1827–1835

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.