Ad Hoc Networks 165 (2024) 103641

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Load-adaptive MAC protocol for frontier detection in Underwater Mobile Sensor Network

Ansa Shermin S. [a,*], Bhavya Mehta [a], Sarang C. Dhongdi [a], Mandar A. Chitre [b]

[a] Department of EEE, BITS-Pilani - K K Birla Goa Campus, Goa, India

[b] Department of Electrical and Computer Engineering, and ARL, Tropical Marine Science Institute, National University of Singapore, Singapore

ARTICLE INFO

Keywords:
Underwater Mobile Sensor Network (UWMSN)
MAC
Autonomous Underwater Vehicle (AUV)
Autonomous Surface Vehicle (ASV)
Frontier detection
Underwater phenomena
Ranging/Navigation

ABSTRACT

This work proposes a load-adaptive Medium Access Control (MAC) protocol for the frontier/boundary detection application of underwater phenomena using Underwater Mobile Sensor Network (UWMSN). A leader-follower architecture of a swarm of underwater vehicles is proposed here. Autonomous Underwater Vehicles (AUVs) traverse a random mobility pattern beneath one Autonomous Surface Vehicle (ASV) (leader) in the proposed network. ASV has to guide multiple-follower AUVs in the event of interest. The vehicular swarm aims to explore the frontiers in the event to build the map. Load-adaptive MAC protocol is therefore proposed and implemented in this hybrid multi-vehicular network to ensure seamless vehicular communications. The ASV has navigational capabilities to aid the AUVs in navigation and data collection. The proposed MAC protocol can adjust the dynamic mobility and load in the network. The protocol aims to provide dynamic Time Division Multiple Access (TDMA) slots for the AUVs wirelessly linked in the vicinity of the ASV. These slots are used for ranging/navigation and data transmission. Additional urgent data from any AUVs can be transmitted in open Carrier Sense Multiple Access (CSMA) protocol following the TDMA duration. Results have been generated by comparing protocols like CSMA, ALOHA, and TDMA with the proposed Load-Adaptive MAC protocol. The protocols have been compared to the throughput vs number of nodes and throughput vs simulation time. It has been observed that the proposed MAC can perform better than ALOHA and CSMA protocols. Nevertheless, it can produce comparable results for TDMA protocol while supporting the dynamic mobility and load in the network meantime supporting urgent data transmission for nodes in demand.

1. Introduction

In the past decade, there has been tremendous growth in underwater explorations, from static mooring systems to Underwater Mobile Sensor Networks (UWMSN) [1]. Due to their mobility, in much faster ways, these networks can explore the oceans, from underwater monitoring and surveillance to ocean safeguarding purposes [2]. Nevertheless, these missions must make trade-offs between sensing, coverage, and resolution for efficient functioning, especially considering large-scale spatio-temporal phenomena like underwater plumes.

The UWMSN network, consisting of multiple autonomous underwater vehicles (AUVs), should cooperate for large-scale phenomena exploration missions. There are certain aspects to consider while building these networks, such as their self-organization abilities, long-term mission capabilities, and unknown underwater environments. So, one must focus on autonomy, sensing, endurance, and communication to build an effective network. For instance, assume that these underwater vehicles are employed in a cooperative mission, say, frontier exploration, a widely explored domain for multi-robotic application [3].

Cooperative autonomy and communication are vital in improving the robotic networks' mission performance, especially in large-scale applications like plume monitoring. Researchers have developed approaches like adaptive sampling techniques [4] to monitor these events. Based on the gathered data, these strategies aid in the planning and executing missions to guide the AUVs to the Region of Interest (RoI) and produce the most information from it [5]. Generally, these sampling techniques use onboard algorithms to determine source localization, boundary tracking, or area mapping for the event of interest. Initially, the AUVs focus more on area coverage goals, such as patrolling or territory exploration [6]. Whenever finding an event is highly likely, they shift to the event detection mode [7] and then track the event based on the desired classification algorithms.

Many works have been conducted to build efficient networks comprising several AUVs to learn about these phenomena's geographical and temporal variations. The necessity of multi-vehicle communication and the networking protocols involved at several layers, including

* Corresponding author.
E-mail address: p20180017@goa.bits-pilani.ac.in (Ansa Shermin S.).

https://doi.org/10.1016/j.adhoc.2024.103641
Received 24 May 2024; Received in revised form 6 August 2024; Accepted 26 August 2024
Available online 31 August 2024
1570-8705/© 2024 Elsevier B.V. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

the physical, data link, and network layers, are inevitable in developing such networks. In addition to that, georeferencing the sample locations is necessary for successful autonomous robotic missions. For the precision of collected data, the location information is therefore important.

Ranging/localization techniques help with the georeferencing of samples. Global positioning system (GPS) is the most widely used positioning method to solve navigational problems in both terrestrial and aerial robots. However, the electromagnetic waves conveying the positioning information cannot adequately penetrate the water column, and AUVs cannot benefit from GPS localization [8]. Therefore, it is essential to employ a cooperative navigation/ranging technique that could help in the navigation of the vehicles involved in the mission. The network should also act as the support system for georeferencing the AUVs.

This work has developed a topology using multiple AUVs that could imbibe the autonomy and communication of an event monitoring mission. These AUVs are engaged in frontier or boundary detection of an underwater phenomenon. AUVs explore the frontiers in random walk mobility pattern to build the map and send the sensed data to ASVs at the surface. The ASVs have to further send the data to a mother ship, the central controller for building the map. The network's primary goals for large-scale applications like plume monitoring are to provide efficient communication and ranging/navigation of multi-AUVs for data collection missions. Therefore, this work proposes a MAC protocol for a swarm of AUVs traversing a random walk mobility pattern with one leader ASV guiding multiple-follower AUVs. For the frontier/ boundary detection application described here, the AUVs have to explore the local area beneath the ASVs and follow the ASVs to new areas in the environment. The AUVs have to send the sensed data to the ASV, which has to be used to update the map of the phenomena.

Various approaches are available to plan an underwater mission. The mission can be priorly planned, and the AUVs or the agents have to perform some scripted actions that are purely deterministic in nature. These mission types fall under the model-driven approaches [9], where the highly predictable missions lack decision-making capability, resulting in very little vehicle autonomy.

Since the chosen application needs the vehicles to react to the dynamic environment during their exploration, a model-driven approach that considers deterministic mobility cannot be feasible. In these cases, the missions must consider data-driven approaches with non-deterministic mobility for mission planning. This gives the vehicles the liberty to make decisions that consider environmental cues even without curtailing the decisions from the central controller in the architecture.

Therefore, a suitable MAC protocol should be proposed to connect autonomy and communication based on the particular application. Many protocols for underwater networks are available in the literature, from contention-based and schedule-based MAC protocols [10–12]. These protocols are not suitable for networks with dynamic loads. However, combining these MAC protocols to form a hybrid MAC can address the dynamic load in the network. Although there are many hybrid MAC protocols available in the literature [13,14], they all are developed for different topologies and cannot be applied to the required network topology; therefore, this work proposes a hybrid MAC protocol named Load-Adaptive MAC protocol that could adhere to the required application and network objectives.

The design of MAC protocol in an underwater scenario depends on the innate characteristics of the underwater environment, such as high propagation delay and longer communication distance between the vehicles. This creates a situation where the vehicles are only sometimes within the communication radius of the ASV (leader). In the proposed topology, the ASV has to allot the slots dynamically concerning the varying number of vehicles to have effective communication among them. Developing a hybrid MAC protocol that could adapt to varying loads in the network is a suitable solution. A load-adaptive MAC protocol is henceforth proposed in this work. The protocol aims to provide dynamic TDMA slots for the AUVs wirelessly linked/in the vicinity of the ASV. These slots are used for ranging/navigation and data transmission. Additional urgent data, if available, can be transmitted in open CSMA protocol following the TDMA duration.

The paper is organized as follows. Section 2 gives the related works. Network topology is described in Section 3. The exploration strategy for building the map is described in Section 4. Section 5 describes a Load-Adaptive MAC protocol for frontier detection. Section 6 demonstrates the proposed protocols of UWMSN using the UnetStack Simulator. Finally, Section 7 concludes the paper.

2. Related works

This paper focuses on the cooperative mission of mobile underwater networks. The nodes/AUVs must work together to complete missions by sensing, data gathering, and sharing information. These nodes should communicate with each other in a cooperative mission where sharing data is essential. Nodes, for example, need to transmit timing information, location data and sensed data during various stages of the mission. Nodes in network operations may need neighbouring locations to map the survey area, prevent vehicle collisions, or serve as relays between nodes. Therefore, the effective utilization of the shared underwater channel is necessary.

The data link layer's MAC layer enables the various nodes in a network to use a shared channel efficiently. However, due to the dynamic underwater environment, the underwater MAC protocol requires more attention than its terrestrial counterpart. Apart from inherent underwater characteristics like constrained bandwidth, significant propagation delays, high bit error rates, loss of connectivity, and multipath effects, UWMSN has one more factor to consider: mobility. Therefore, the design of MAC protocols should account for mobility in underwater scenarios [15,16] by addressing the spatio-temporal variability of the dynamic underwater environment.

The MAC protocols for the UWMSNs are classified into contention-based, scheduled-based, and hybrid MAC protocols.

1. Contention-based MAC
   Many works in the literature consider the contention-based MAC protocol for the UWMSN. One such example is the Cooperative Target Detection (CTD), described in [10]. A random access and handshake-based protocol is applied to this network, and the performance is evaluated. Distance Aware Collision Avoidance Protocol (DACAP) with handshake mechanism is compared with the CSMA ALOHA protocol, a basic random access technique [17]. DACAP protocol is an advanced version proposed for a dynamic network compared to the basic random access counterpart, and the transition is described below.
   Considering the CSMA ALOHA, the channel has to be sensed before data transmission [18]. If transmission occurs, the node will detect the transmission, back off, and resume after an arbitrary amount of time, but this method is more prone to packet collisions. Therefore, a handshake mechanism is introduced, and the protocol is modified as Multiple Access for Collision Avoidance protocol (MACA) [19]. Control packets, RTS and CTS (Ready To Send and Clear To Send) are used for the handshake mechanism. However, the distance between the nodes has to be proportional. To cope with the varying distance, a carrier sense mechanism is introduced into MACA [20], known as Floor Acquisition Multiple Access protocol (FAMA). This protocol is further modified for a dynamic AUV network [16] and known as Slotted FAMA (SFAMA). The time slotting technique of SFAMA allows the node to send data only at the start of the time slot. It prevents multiple concurrent transmissions, eliminating the option to select longer control packets.

Whereas, by enabling a node to use varying handshake lengths for different receivers, the Distance-Aware Collision Avoidance Protocol (DACAP), as proposed in [21], minimizes the average handshake duration and thereby reduces the frequency of packet collisions.
A traffic-aware protocol called Load Adaptive CSMA/CA MAC (LACCM) for single-hop networks is described in [22], where the static sensor node serves as relays for one or more AUVs. This protocol uses RTS and CTS packets based on a handshake mechanism. The node delivers two data packets in heavy load traffic with a single handshake to minimize the overhead caused by the control packets.
The nodes in contention-based MAC protocols have to compete for the channel to send data, which is suitable for a network with smaller network traffic as it incurs only a small end-to-end delay. However, as the network traffic increases, the competition for channel access increases, leading to higher end-to-end delays.
For UWMSN, current data transfers are rather important than transmitting large packet sizes or data packet trains. The dense AUV network configurations are highly interconnected, leading to high traffic; scheduled-based MAC protocols can support packet generation and data transfers.

2. Scheduled-based MAC
TDMA is the most commonly used scheduled MAC protocol for AUV networks. TDMA-based MAC protocols are often known for simplicity, fairness, and energy efficiency. Collisions, idle listening, and over-hearing can be avoided in these protocols [23], ensuring the successful reception of data packets without collision. Due to the larger guard bands used in this conventional TDMA, the protocol is suitable for a smaller AUV network.
However, the larger guard bands in conventional TDMA protocols effectively utilized the concepts of spatial reuse in [15]. Whereas concurrent transmissions are allowed in [24], given that no collisions are occurring at the receivers and at the same time considering the possible interference from neighbour nodes.
Many centralized AUV networks use the TDMA MAC protocol [11,12,25,26] in which communication is managed by a buoy or sink at the surface. According to the TDMA schedule, communication is only between the AUV and each sink node separately.
3. Hybrid MAC Regarding the proposed network topology, there will be a dynamic distribution of load in the network; neither of the two classifications is, therefore, unsuitable. However, certain MAC protocols are developed under the hybrid MAC category whose combination can take advantage of contention-based and scheduled-based MAC protocols [13,14,27].
A hybrid MAC protocol can address the network traffic load's non-uniform distribution. Based on the network topology, different protocols are combined to adapt to the network modes. A Cluster-based TDMA with (code division multiple access) CDMA protocol described in [28] combines TDMA and CDMA protocols. This clustering-based scheme is scalable and capable of supporting the spatial reuse of channel resources. The mobile entities form the cluster where the inter-cluster communication is achieved through TDMA, and adjacent clusters use the CDMA technique. Another example is DCO-MAC, explained in [13], where the network is partitioned into two-tier subnetworks based on the network traffic. A contention-based MAC is used in a subnet with lesser traffic, and a contention-free MAC is in the network with a higher traffic load. Nevertheless, with reduced end-to-end delay, the network is prone to interference between the corresponding subnetworks, diminishing the network throughput.
Another hybrid MAC protocol named load-based slot allocation (LBTSA) is proposed in [14], which can adapt well to a dynamic network with varying competitors/load. This protocol is designed to adapt to varying network loads by switching to the access mode at any time according to load changes. TDMA and carrier sense multiple access with collision avoidance (CSMA/CA) is used here as the two modes.
Despite the hybrid MAC protocols described above, the proposed MAC protocol in this work has to be designed under the specific application requirements subjected to the mission objectives. The topology proposed in this paper is developed for an application-specific scenario, allowing georeferencing of the collected samples and AUVs. A hybrid MAC protocol named Load-Adaptive MAC combining TDMA and CSMA MAC protocol is developed in this work.
It has been observed that much work has been developed so far is solely for ranging/navigation of the AUVs with the aid of an ASV [29–31]. The CSMA and TDMA MAC protocols are commonly used in these networks for sending ranging/navigation information. However, most of the work has opted for TDMA-based scheduling to ensure reliability in communication. It is worth mentioning that vehicles/nodes in these networks use synchronized and high-stable clocks on each ASV and AUV board, and the ASV transmits navigation packets within a predetermined interval [32] based on the required precision needed.
The aim of the proposed work is frontier detection applications where georeferencing the collected samples is necessary. The MAC protocol proposed here should also better fit in a dynamic underwater environment where the AUVs are in constant motion, and the number of competitors constantly changes. Therefore, the MAC protocol developed should accommodate varying competitors or loads. The proposed MAC has, therefore, developed a dynamic TDMA for data collection and ranging, which should be adaptive to the number of AUVs joining the ASV during the neighbour discovery process and can guarantee the TDMA's reliability. The protocol also accommodates the need to send additional data with the help of the CSMA MAC protocol, which could help improve the network's flexibility.
The following subsection will give some insights into how the network topology has been developed and the purpose of the network formation.

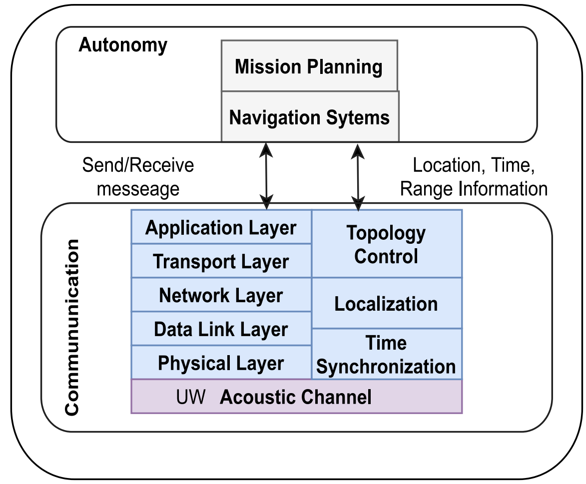


Fig. 1. Relationship between autonomy and communication.

2.1. Concept of autonomy and communication in underwater vehicles

Underwater networks for collaborative networks should balance autonomy and communication to ensure mission success. In a dynamic underwater environment, it is paramount to determine strategies for navigating, adapting, and working toward the network objectives.

There are many approaches to carrying out an underwater mission. In a model-driven approach, the mission is priorly planned with a sequence of scripted actions [33]. These missions are carried out with little autonomy, following the priorly planned actions, and are executed one after the other. This approach uses a world model for the mission execution [34].

Various examples exist for these pre-deployment missions for large-scale applications, [9] where the vehicles follow pre-defined paths and are strictly constrained to restricted spaces. One such example is monitoring the hydrothermal vent plumes in a large area of 400 km for 4 days; this is a multi-phase operation, where the subsequent dives were planned following the outcome of previous dives [35].

In this case, the high level of mission predictability results in almost no decision-making capabilities for the vehicles. Despite going with the operational strategy of the model-driven approach, combining a reactive component that enables the vehicles to react in response to the changing environment during the exploration is desirable.

Therefore, data-driven policies can be used in an environment where communication with the central controller is not necessary to be always possible. Using these approaches, the vehicles can plan and optimize their paths based on the collected information [33,36], or there is another intermediate option for the AUVs to divert from the preplanned trajectories concerning the collected data.

Nevertheless, in reactive approach where communication between the AUV and the central mission operator is not possible, the mission should rely solely on the vehicle's decision-making abilities. The spectrum of autonomy architectures provides a range of options, from purely reactive policies to deliberative solutions. Reactive or behavioural architectures define the mission as a sequence of pre-defined behaviours. These behaviours are triggered by external stimuli and mission objectives, following the sense-react principle [37]. In purely reactive architectures, a world model is not employed, and actions are determined solely by the data collected by the AUV. This approach proves effective in dynamic environments with limited computational requirements.

On the other hand, deliberative architectures involve planning that balances near-term objectives with long-term or dynamically changing goals. Deliberative approaches rely on a world model, incorporating acquired information with a priori knowledge of the scenario. This integration enables AUVs to make more informed decisions, considering current sensor data and a broader understanding of the environment [8].

The choice between reactive and deliberative architectures depends on the specific mission requirements, environmental characteristics, and computational resources available. Reactive architectures are often favoured for their simplicity and effectiveness in dynamic environments, while deliberative architectures offer more sophisticated decision-making capabilities through world models.

Fig. 1 shows the relationship between autonomy and communication. It has been observed that the mission planning and navigation systems can be controlled based on the message transmission and reception of messages from the communication module and sharing the location, range information and time sync information. The proposed work enables a ranging/navigation-enabled network architecture and provides the slots for communicating the ranging information. Thus enabling the network to bind autonomy and communication.

3. Network topology

This work develops a topology formation of one-hop multi-AUV networks for event monitoring applications. A Load-Adaptive MAC protocol comprising CSMA and TDMA protocol is proposed here for a network with ASVs at the surface and AUVs underneath. The TDMA MAC sends the Data and ranging information of AUVs to the ASVs,

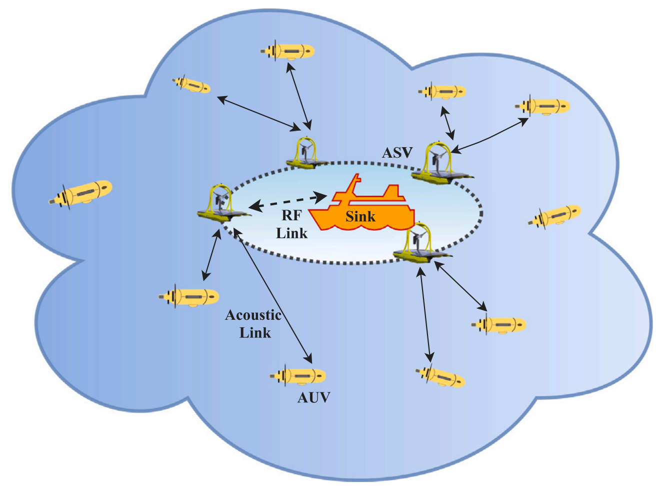


Fig. 2. A cooperative Underwater Mobile Sensor Network.

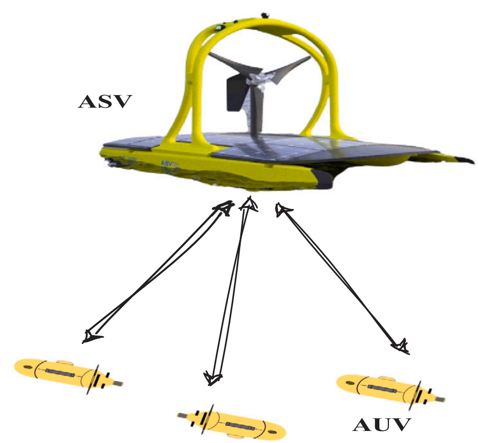


Fig. 3. Network Topology of UWMSN for boundary/frontier detection.

while the contention MAC can send additional information or any urgent messages by the AUVs to the ASVs.

The UWMSN network, consisting of multiple AUVs and ASVs, should cooperate to explore the frontiers for building the map of the ocean phenomena. Fig. 2 shows a network of UWMSN. In this network, a Central Controller (CC) at the surface, which is a mother ship/sink, collects the data from the ASVs to build the map. The ASVs act as leaders to the AUVs. The AUVs move in random mobility patterns concerning the environmental cues based on the sensed data.

The data must be collected first by the AUVs and sent to the ASVs. Fig. 2 shows multiple sub-networks comprising one ASV and multiple AUVs. Therefore, the AUVs have to send the data to the ASV in the sub-network in which it is the current member. The proposed network is shown in Fig. 3. There is one ASV with a GPS having an ideal clock at the surface, and multiple AUVs are one hop away from the ASV. This work aims to build a network for data collection with ranging capabilities.

This paper speaks about a topology developed using multiple AUVs that could imbibe the autonomy and communication of an event monitoring mission. The network should, therefore, be able to form an autonomy-based networking system. In this network, the ASV helps in the ranging/navigation of the AUVs. Communication is enabled here for both ranging and data collection purposes. MAC protocols can be effectively used for message exchanges for data collection and the range measurements among the vehicles [38]. It is more important to design a

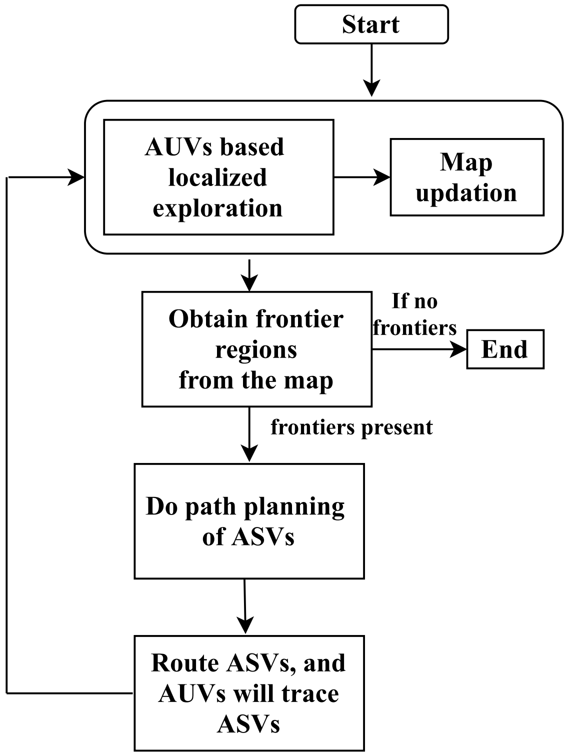


Fig. 4. Flow Chart for frontier detection.

suitable MAC protocol that helps to impart effective channel utilization for these vehicles [39]. Reliable data transmission is the primary focus for these large spatio-temporal applications assigned for data collection missions.

Similar to the MAC protocols for data collection, many works have been conducted to evaluate network-based localization performance. Two MAC approaches, TDMA and CSMA, are proposed in [40]. An interval analysis-based navigation system is used here for periodic localization/ranging. Determining a feasible interval for ranging/localizing the network based on the required precision for the application would result in less message overhead. It would ensure the proper working of all the AUVs in the network, especially while using a TDMA MAC, as it can ensure a high packet delivery ratio.

The proposed MAC combines TDMA for data collection and ranging and CSMA MAC to send urgent messages. The MAC has to work dynamically to the network requirements. The MAC protocol designed here is adaptive to the number of AUVs joining the ASV during the neighbour discovery process. It further provides the reliability of TDMA in the data collection/ranging phase. It ensures the flexibility of the CSMA protocol when sending additional information to the AUVs in need. CSMA can also help to include variable load/AUVs in a cycle.

4. Exploration strategy for building the map

The AUVs have to explore the region of interest for building the map. For precise data collection, the AUVs must have their position estimates, especially when collecting large amounts of data from the event of interest. As shown in Fig. 2, ASVs are leaders for multiple AUVs. The AUVs move in random motions to gather information.

The main objective of the network is to map the event. There is one central controller or the mother ship that has some prior knowledge about the environment to be mapped and stays hovering in that region. The vehicles are grouped into two groups in this proposed strategy. There are landmark vehicles and mapper vehicles. The landmark vehicles are the ASVs equipped with GPS, and the mapper vehicles are AUVs, whose job is data collection from the RoI. The ASVs help to direct the AUVs to the event location.

Fig. 4 shows the flow chart of how the event is being mapped. The mapping of an event is an iterative process [3]. With each iteration, the AUVs need to map the local region surrounded by the ASV. Then, the data has to be sent to the ASVs, and the ASVs have further forwarded it to the CC based on the information collected by the AUV; the CC will update the map.

This strategy helps to prioritize exploring and mapping unknown areas in the environment. Frontiers are the regions between the explored and unexplored regions. This adopted strategy seems very effective as there are two sets of vehicles to direct the fleet toward an area in the environment and the other to acquire measurements to build the map of that area. The ASV will direct the fleet toward the event's region, maximizing the potential for new information to be added to the map.

While exploring specific frontier points, the ASVs are assumed to be in the quasi-static state. After exploring those points, the ASV has to move to the next position, and the AUVs must follow them. The network is assumed to be initially positioned at a fixed location. AUVs must continually send messages that include the range measurements and their local environment to the ASVs.

5. Load-adaptive MAC protocol for frontier detection

Information exchanges among the vehicles are indispensable for the seamless connection of a cooperative network of marine vehicles. Developing an ad-hoc network of these vehicles by enabling acoustic communication is challenging. There are certain factors to look at while designing these networks, such as the design of the MAC layer, which is responsible for managing the shared medium for communication among these vehicles to lower packet collisions.

In a dynamic underwater environment where the AUVs are always in constant motion, there is always a change in the number of competitors. Considering the topology mentioned above, it is not less than the case. Therefore, the MAC protocol developed should accommodate varying competitors or loads. Since the application under consideration is a data collection mission, georeferencing the samples helps increase the precision of the collected data.

In an ad-hoc network of AUVs with acoustic modems, these networks can use their possibility to cooperate for navigation and localization rather than relying on expensive sensors for navigation purposes. The topology mentioned here comprises heterogeneous vehicles with variable capabilities. The ASV acts as the master node and has the GPS fitted to it, helping to localize the AUVs in its vicinity. This method can leverage positioning services without deploying dedicated transponders and acoustic systems but use the same underwater acoustic modems adopted for data transmissions in the network.

The topology for the network is shown in Fig. 3. There is an ASV at the surface, and multiple AUVs one hop away from the ASV. The ASV will act as a ranging aid for the multiple AUVs. This is a centralized network with a star topology. There is one central coordinator node, the ASV, with N slave nodes, i.e., AUVs.

While developing a MAC protocol for the UWMSN, we need to consider its adaptability to the spatio-temporal variations of the desired applications. Moreover, it is of utmost importance for the protocol to ensure the reliability of data transfer to the destined nodes with lower collisions. Various MAC protocols for UWMSN are in the literature. Few are under contention-based MAC, which uses CSMA protocol, and few others in scheduled based MAC propose using TDMA-based MAC protocols.

Certain assumptions are made when building a communication protocol for these vehicles. Furthermore, this work has to scale up while developing the communication and coordination strategies for the ASVs, acting as the landmark vehicles for the topology. Rather, a single ASV and multi-AUV topology are considered here for developing

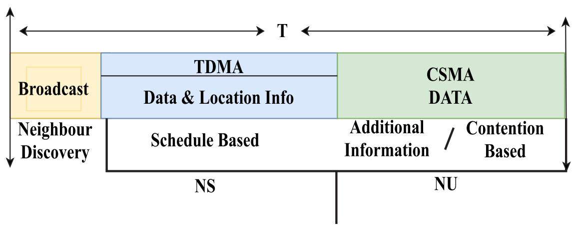


Fig. 5. Frame Structure for the Load-Adaptive MAC.

Algorithm 1: Load-Adaptive MAC Protocol Pseudocode for ASVs

Input : *Network parameters, ASV parameters*
Output: *collecting Data from AUVs using Load-Adaptive MAC*

```
Start();
ASV wait_for_some_time;
ASV initiate_neighbor_discovery;
broadcast_initialization_ msg(Delay_Length,
  TDMA_Slot_Length, CSMA_Slot_Length);
Protocol(INIT);
Increment_Neighbour_TX_Count( NBTX++);
wait_for_acknowledgements_from_AUVs;
if Acknowledgement(received) then
    for each ACK do
        decode node _address;
        update the node_address_list;
    end
    sort the node_address_list in ascending order;
    increment _Neighbour_RX count (NBRX++);
end
Initiate_TDMA_Phase;
Broadcast_TDMA_Initiation_Msg(TDMA_Time_Slot_Info);
Increment_Count( TDMATX++);
wait_for_TDMA_Msg;
if TDMAmsg(received) then
    for each Msg do
        decode _location_ and sensed_data;
    end
    collect the ranging and sensed data info;
    increment count (TDMARX++);
end
Initiate_CSMA;
CSMA Phase Duration =
  TDMA_slot_time × _Size_of_Neighbour_List;
if CSMAmsg(received) then
    for each Msg do
        decode_urgent_data_info;
    end
    collect the urgent_data info;
    increment count (CSMARX++);
end
Stop();
wait_for_next_broadcast;
```

Algorithm 2: Load-Adaptive MAC Protocol Pseudocode for AUVs

Input: *Network parameters, AUV parameters*
Output: *Data Collection by AUVs via Load-Adaptive MAC*

```
Start();
Initialize_AUVs;
while Msg_Received() do
    process_Message_(msg);
    if msg_protocol_INIT then
        Decode_and_ update_ delay_Length _from_ msg;
        Decode_and_update _tdmaSlot _Length_ from_ msg;
        Decode_ and_ update _csma_Slot_Length_ from _msg;
    end
    Increment_count (NBRX++);
end
if channel_Status_busy() then
    Calculate_backoff_ Time;
    Send_ ACK_protocol_msg (Node_ Address);
    Increment_count (NBTX++);
end
else
    Send_ ACK_protocol_msg (Node_ Address);
    Increment_count (NBTX++);
end
while Msg_Received() do
    if msg_protocol _(TDMA_INIT) then
        Decode_TDMA time_slot_allotment;
        Increment_count_TDMARX++;
    end
end
Schedule_Transmission_(msg);
TDMA_time _slot_(current_ node);
Encode_position_data_(x,y,z)_and_sense data_PDU;
Transmit_PDU_(msg);
Increment_TDMATx_count;
Switch_to_CSMA_for_urgent_ TX();
Compute a Random_sense_time();
if Urgent Message detected then
    Send_Msg_via_CSMA_mode;
    Increment_Count(CSMATX++);
    if channel(busy) then
        back-off_some time;
        CSMA_Transmit_(msg);
        Increment_Count(CSMATX++)
    end
end
Wait_for_next_ INIT(msg) from ASV;
Stop;
```

the MAC protocol. Considering the mobility of the network, the AUVs are in random motion, and it is assumed that the topology, i.e., the no. of nodes, is not changing during a cycle.

However, regarding the proposed network, the protocol design should consider the network's dynamic requirements by accommodating varying loads. Therefore, a Load-Adaptive MAC protocol is proposed in this paper that uses the combination of these schemes to allocate/share underwater channels. The Load-Adaptive MAC protocol could combine the virtues of both contention-based MAC and scheduled-based MAC. This can be attained by utilizing the reliability of a schedule-based MAC and the flexibility of a contention-based MAC. TDMA MAC can exchange sensitive information or information with higher priority; here, it is considered ranging information with sensed data. On the other hand, CSMA MAC can add flexibility to the network by giving additional slots to send urgent messages. Since, in the network, the nodes move randomly, they may move from one ASV to another, or there would be some substantial changes in their location, making the static TDMA a non-viable choice. The Load-Adaptive MAC design supports a dynamic network, allowing the liberty of variable load (by varying the no. of AUVs).

There are three phases for the Load-Adaptive MAC protocol: (i) the neighbour discovery phase, (ii) the TDMA phase for ranging and data collection, and finally, (iii) the CSMA phase for Urgent data transmission. Fig. 5 shows the proposed protocol's frame structure. After the neighbour discovery process, Ns slots will be dedicated to the scheduled MAC and Nu slots for the unscheduled MAC. The Ns scheduled slots use TDMA MAC for performing the high-priority transmissions, here, for sending the sensed data and range information of AUVs to the ASV [13,24], and Nu slots are dedicated to the contention-based MAC for the transmission of urgent messages.

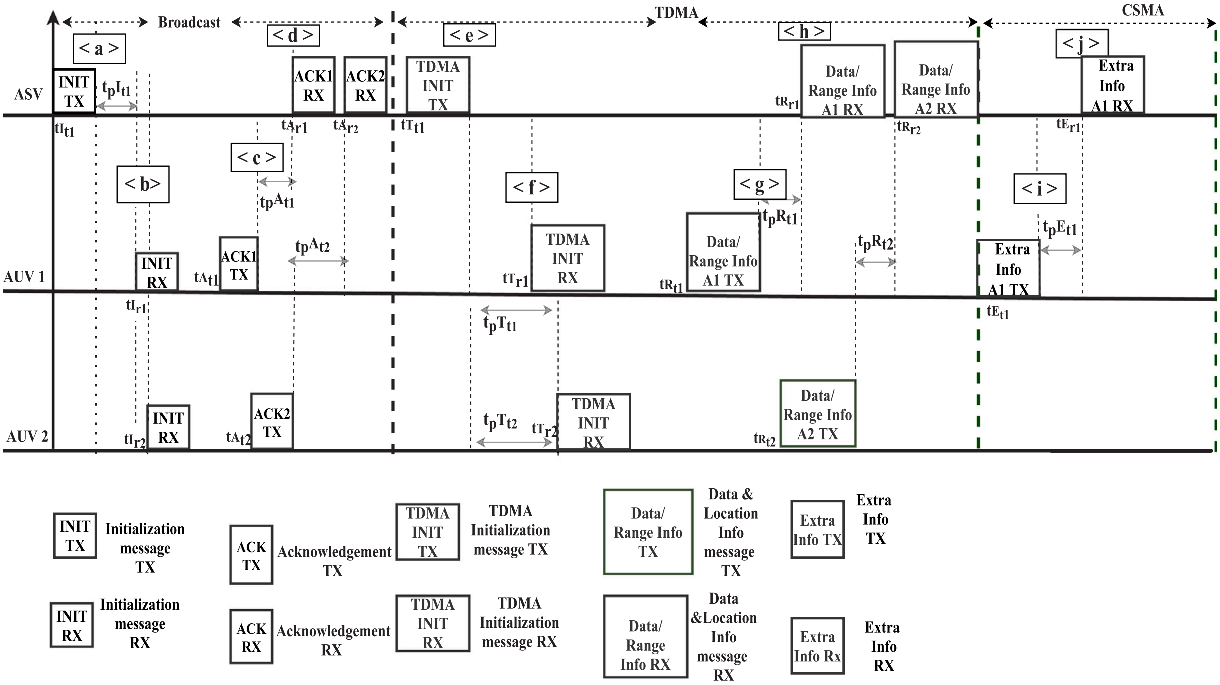


Fig. 6. Graphical representation of message exchanges by ASVs and AUVs.

Though the network has only considered a single ASV with multiple AUV followers, since the AUVs are in motion and the network as a whole has multiple subnetworks, the AUVs are prone to move from one subnetwork to the other, making variable load for the ASV. Therefore, neighbour discovery in each cycle would help to adjust with the variable load, and all the AUVs accommodated in the current subnet would get the opportunity to send their data to their corresponding leader ASV.

5.1. Neighbour discovery phase

The graphical representation of the Load-Adaptive MAC is shown in Fig. 6. It shows the message exchanges in three phases. In phase one, the neighbour discovery phase, messages are transmitted via a broadcast mechanism. The message exchanges for this phase are denoted as the process from $< a >$ to $< d >$. $< a >$ represents the broadcast of the initialization message by ASV at time tI_{t1}. The message contains the TDMA slot length, CSMA slot length and the delay length. The message is received by AUVs 1 and 2 at tI_{r1} and tI_{r2}, respectively, after their corresponding propagation delay t_pI_{t1}. The AUVs will decode and update the TDMA slot length, CSMA slot length and delay length (neighbour discovery duration) from the message. The process $< c >$ represents the transmission of the acknowledgement message by the AUVs at tA_{t1} and tA_{t2}. This contains their corresponding ID. If the AUV finds the channel busy while sending the ACK message, it will back off for a random duration and then transmit the message while the channel is not busy. After the duration t_pA_{t1} and t_pA_{t2} the message is received by the ASV at tA_{r1} and tA_{r2}, in the process $< d >$. The ASV will decode the message to find the node address and sort the message in ascending order.

5.2. TDMA for ranging and data collection phase

The ASV waits for some time before starting the TDMA phase. The process represents the TDMA phase from $< e >$ to $< h >$. During the process $< e >$, the ASV will send the TDMA initialization message at time tT_{t1}, and it will be received by AUV 1 and 2 after the propagation delay of t_pT_{t1} and t_pT_{t2} represented by the process $< f >$. The ASV's initialization message contains the order in which the message has to be sent. The AUVs will send the ranging and data packet during their corresponding TDMA slot, at tR_{t1} and tR_{t2} during the process <g>, the ASV will decode the location and sense data from the received packets at tR_{r1} and tR_{r2} in the process <h>.

5.3. CSMA for urgent data transmission phase

The next phase is the CSMA phase, the process from $< i >$ *to* $< j >$. This phase lasts for a duration equal to the TDMA slot duration multiplied by the size of the neighbour list. The AUVs with a message to send are sending it at time tE_{t1}. This is represented as the process $< i >$. The ASV at $< j >$ receives data after the propagation delay at tE_{r1} and decodes the data.

5.4. Algorithms for load-adaptive MAC protocols

The algorithm for a Load-Adaptive MAC cycle is described as two separate processes running in the ASV and AUVs. Algorithm 1 explains how the Load-Adaptive MAC protocol is implemented in the ASV. It gives an overview of the process happening in the ASV, for instance, the process described in a graphical representation in Fig. 6 such as $< a, d, e, h, j >$. These processes are explained in accordance with Fig. 6. The pseudocode starts from the ASV initializing the neighbour discovery process after an initial wait time. The ASV broadcasts the initialization message and waits for the ACK message from the AUVs. After receiving the messages, the ASV will decode the node address and sort them in ascending order. Further, it initiates the TDMA phase by broadcasting the message containing the slots for each AUV to transmit. The slots are assigned in the order of node address. After collecting the ranging and sensing information from each AUV, the AUV's location and sensed information are decoded. The ASV waits for any urgent data in the CSMA phase for a duration equivalent to the TDMA slot duration multiplied by the number of neighbours.

The pseudocode for the AUV in the Load-Adaptive MAC is given in Algorithm 2. The process $< b, c, f, g, i >$ is represented in the graphical representation of Fig. 6 describing steps carried out by the AUVs. While receiving the initialization message from the ASV, the AUVs will decode

```
import org.arl.fjage.*
import org.arl.unet.*


    @Override
    void startup() {
        //This method is called just after the stack is running
        // look up other agents and services here, as needed
        //Subscribe to topics of interest to get notifications
    }

    @Override
    void processmessage(Message msg) {
        // process other messages, such as notifications here
        //If a message is not important, it can be safely just ignored
    }
}
```

Fig. 7. Basic structure of an agent.

```
1.ASV startup()

- Subscribe to the DATAGRAM service to listen for incoming messages.
- Initialize phy agent for physical layer communication.
- If phy is null, attempt to initialise phy again.
- Subscribe to the topic so the phy agent can receive messages from the physical layer
    .
- Add WakerBehavior to trigger periodic neighborBroadcast() for neighbor discovery.
- tdmaBroadcast() is triggered after the nodes are acknowledged.

2.AUV startup()

- Initialize a random number generator (rnd) for randomisation.
- Subscribe to the DATAGRAM service to listen for incoming messages.
- Initialize phy agent for physical layer communication.
- Initialize node agent to obtain node information.
- Subscribe to the topic so the phy agent can receive messages from the physical layer
    .
```

Fig. 8. Functionalities of startup() message created for the agents ASV and AUV.

and update the parameters like the TDMA slot length, CSMA duration, and delay length. Then, the AUVs will send the ACK message with its node address. If the channel is busy, it waits for some time before the next transmission. Then, after receiving the TDMA initialization message from the ASV, the AUV has to encode its location and sensed data information and send it to the assigned slot. If the AUVs have further data, they must send it to the ASV via CSMA protocol.

6. Demonstration of proposed protocols of UWMSN using UnetStack simulator

The above protocol is implemented in the UnetStack simulator [41] developed by ARL. The Unet framework provides core services, messages, agents, and APIs needed by UnetStack. Unet basic stack is a collection of agents providing services and functionality required by typical Unets. These agents and the Unet framework are sufficient to build fully functional Unets. The Unet premium stack is a collection of agents providing advanced functionality and/or higher performance. The basic structure of the agent developed in this work is provided in Fig. 7.

The creation of an agent involves the following steps:

1. The startup () method looks up all agents providing the service and subscribes to any notifications from any of these agents. When a notification arrives, the processMessage () method will be called.

2. The processmessage () processes and evaluates the message before moving to the next step in the protocol.

So, based on this in the work, two agents are created for the ASV and AUV, the ASVNode agent and AUVNode agent. The agents describe how to generate and process the messages in the ASV and AUV. Agents are interacting with each other through messages. Messages include requests, responses and notifications. Responses are sent on request, while notifications are unsolicited. Agents support parameters that can be used to configure or monitor the agent.

Messages can not only be sent to specific agents but also can be broadcast on a topic. All agents subscribing to a topic receive a message broadcasted on that topic. Unsolicited notifications are usually sent on topics associated with an agent since an agent does not know a priori which other agent might be interested in that notification. A collection of requests, responses, notifications, and parameters that form a cohesive functionality is known as a service.

Figs. 8 and 9 show the functionalities of ASV and AUV node agents during the startup () and processMessage (). Given these two agents, a complete set of functionalities during the setup () and process methods are explained with respect to Figs. 8 and 9.

The startup () phase is similar to turning on various services, while the processmessage () processes and evaluates the message before moving to the next step in the protocol. Various protocols, like the Initialization protocol (Protocol INIT) for network initialization, by broadcasting TDMA slot length, CSMA slot length, Delay length; Acknowledgement protocol (ACK) for sending acknowledgement message containing the node address; TDMA initialization protocol (TDMA-INIT) for broadcasting TDMA schedules; TDMA protocol (TDMA) to send the

```
1.ASV processMessage()
  - If DatagramNtf with ACK protocol is received:
    - Decode acknowledgement data and update the neighbours list.
    - Increment the counter for received acknowledgments (NBRx).
  - If DatagramNtf with TDMA protocol is received:
    - Decode TDMA data and update counters.
    - Increment the counter for TDMA receptions (TDMARx).
    - Process TDMA data.
  - If DatagramNtf with CSMA protocol is received:
    - Process CSMA data.
    - Increment the counter for CSMA receptions (CSMARx).
- Provide access methods for retrieving parameter lists

2.AUV ProcessMessages()
    - If DatagramNtf is received from another node:
    - Set the channel busy flag to indicate the communication channel is busy.
    - Schedule WakerBehavior to reset the channelBusy flag after 1500 milliseconds.
  - If DatagramNtf with INIT protocol is received:
    - Decode initialisation data and extract parameters such as tdmaSlotLength,
      csmaSlotLength, and delayLength.
    - Respond with an acknowledgement (ACK) if the channel is not busy.
    - If the channel is busy, schedule a WakerBehavior for backoff and CSMA
      transmission.
  - If DatagramNtf with TDMA INIT protocol is received:
    - Decode TDMA initialisation data and process it.
    - Set up transmission in assigned TDMA slots.
    - Schedule a WakerBehavior for CSMA transmission after completing TDMA transmission
      .
  - If DatagramNtf with other protocols is received:
    - Process other protocols
```

Fig. 9. Functionalities of processmessage() created for the agents ASV and AUV.

Table 1
Parameters for simulation of proposed protocols on UnetSim.

Sl. no.	Parameters	Value
1	Frequency of operation (f)	25 kHz
2	ASV depth	0 m
3	Bit Rate (BR)	10k bps
4	AUV depth	100 m
5	channel model	Protocol channel model
6	No. of ASV	1
7	No. of AUVs	3
8	Region of operation	1 km × 1 km

data and ranging information; CSMA protocol to send additional/extra information (CSMA) are created.

In the startup() method, ASV and AUV subscribe to DATAGRAM services for incoming messages and initialize physical layer communication by initializing the phy agent. Both agents will subscribe to the topic to receive messages from the physical layer; initializing the node agent will provide the node information. Agents implement most of their functionality with behaviours. The startup() method is called by OneShotBehavior, and the processMessage() method is called from a MessageBehavior; however, these are implicit behaviours created by the UnetAgent base class. Meanwhile, some behaviours are explicitly added. One such case is adding WakerBehavior to trigger periodic neighbour Broadcas for neighbour discovery, for the process to repeat after a specific interval.

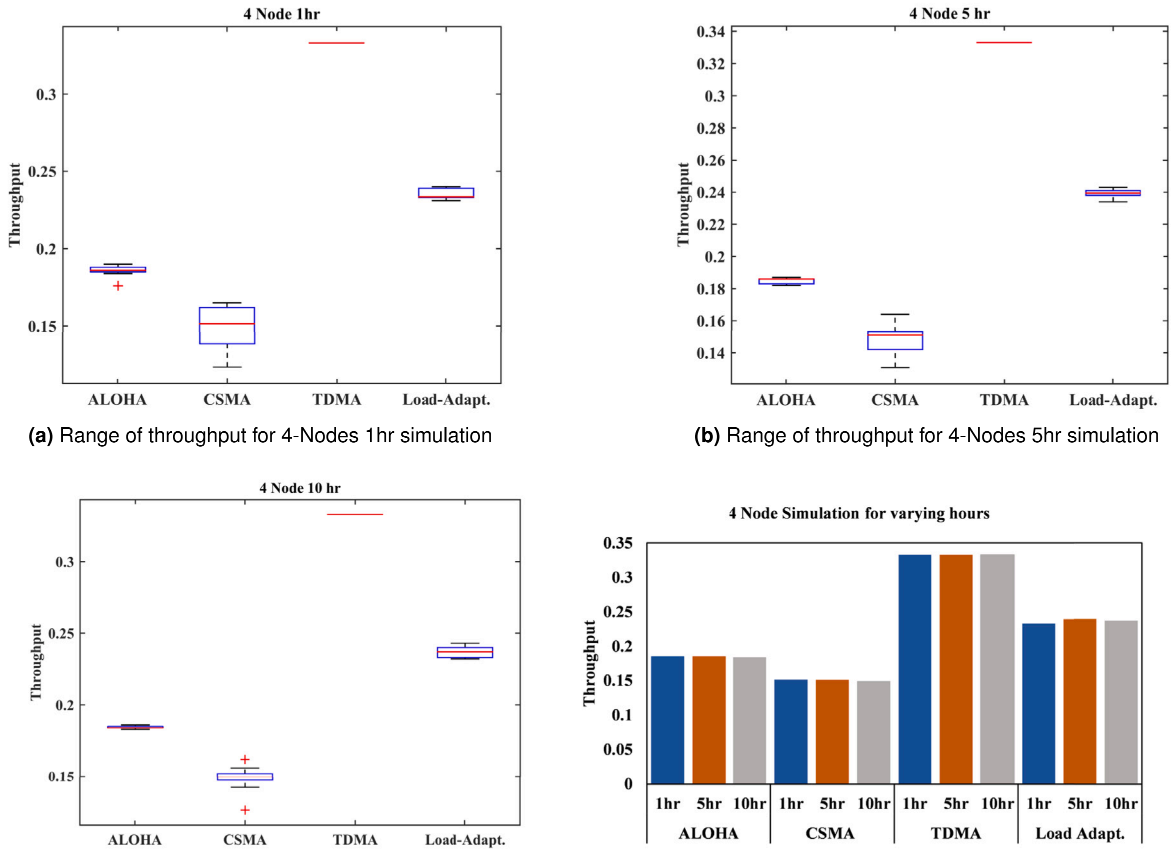
In the processessage () method, the ASV and AUV will process and evaluate the messages received. If the ASV receives a DatagramNtf with ACK protocol, it has to decode acknowledgement data, update the neighbour's list, and increment the counter for received acknowledgements (NBRx). When a DatagramNtf with TDMA protocol is received, it must decode TDMA data, increment the TDMA receptions (TDMARx) counter and process TDMA data. While receiving DatagramNtf with CSMA protocol, it will process CSMA data and increment the counter for CSMA receptions (CSMARx). It will further provide access methods for retrieving parameter lists.

When the AUV receives the DatagramNtf from another node, it will set the channel busy flag to indicate that the communication channel is busy. Then, it schedules WakerBehavior to reset the channelBusy flag after 1500 ms. If it receives DatagramNtf with INIT protocol, it will decode initialization data and extract parameters such as tdmaSlotLength, csmaSlotLength, and delayLength. Then, it will respond with an acknowledgement (ACK) msg if the channel is not busy. If the channel is busy, it will schedule a WakerBehavior for backoff and CSMA transmission. If DatagramNtf with TDMA INIT protocol is received, it must decode TDMA initialization data and process it to set up the transmission in assigned TDMA slots. Again, it has to be scheduled for a behaviour for CSMA transmission after the completion of the TDMA transmission. If DatagramNtf with other protocols is received, it has to Process other protocols.

6.1. Results and discussions

The Load-Adaptive MAC protocol for the proposed topology is compared with other protocols like Aloha, CSMA, and TDMA in the simulation platform. The simulation parameters are given in Table 1. These protocols are compared with respect to the number of nodes vs throughput and simulation time vs throughput.

For a 4-node scenario, simulations are done for varying simulation times. Fig. 10(a), (b), (c) shows the box plots, depicting the range of throughput for the 4-node scenario consisting of 1 leader ASV and 3 AUVs. The box plots are drawn for varying simulation times of 1 h, 5 h and 10 h respectively. From the figures, it is obvious that the range of throughput for respective protocols only shows small variations with respect to the simulation time.



(a) Range of throughput for 4-Nodes 1hr simulation

(b) Range of throughput for 4-Nodes 5hr simulation

(c) Range of throughput for 4-Nodes 10hr simulation

(d) 4-Node Simulation for varying hours

Fig. 10. Box plots showing the range of throughput for 4-Node scenario of various protocols ALOHA, CSMA, TDMA, and Load-Adapt. MAC protocols for varying simulation time and plot of 4-node simulation for varying simulation time.

The time-varying plot of the mentioned protocols in the 4-node scenario is shown in Fig. 10 (d). It has been observed that the throughput of Load-Adaptive MAC is greater than that of CSMA and ALOHA. However, TDMA has higher throughput than that of Load-Adaptive MAC. Nevertheless, TDMA needs fixed schedules and cannot handle the dynamic/varying load in the network. The proposed Load-Adaptive MAC can guarantee the TDMA's reliability and the CSMA's flexibility. The CSMA has the lowest throughput. Combining the TDMA and CSMA protocols can give a better performance comparable to the TDMA MAC protocol. This figure also implies that the throughput of varying simulation hours of 1, 5, and 10 h does not show much deviation in the respective protocols. This ensures the scalability of the protocols under varying simulation times. An example Log File for Load-Adaptive MAC for a 4-node scenario is also included in Appendix.

The protocols have also been tested for varying numbers of nodes. Fig. 11 shows the throughput of various protocols concerning the number of nodes. With varying nodes of 5, 10 and 20 for a constant simulation time of 2 h, Fig. 11(a), (b), (c) gives the box plots showing the range of throughput for various iterations corresponding to the mentioned protocols. To express the relationship between varying numbers of nodes and throughput, median values of iterations are plotted in Fig. 12(a). This implies that throughput does not vary significantly with varying no. of nodes for the mentioned protocols. This implies the scalability of protocols concerning the number of nodes.

Fig. 12(b) shows the packet delivery ratio (PDR) of various protocols to the number of nodes. It has been observed that the PDR of the proposed Load-Adapt. MAC protocol is higher than the CSMA and ALOHA protocols, and results concerning the TDMA MAC are comparable. The PDR of the proposed MAC did not vary much with respect to the number of nodes.

The proposed Load-adaptive MAC shows comparable performance with TDMA MAC at a higher cost of flexibility. The proposed protocol gives scope to the nodes in the network to adapt to the dynamic load and give room for urgent message exchanges when demanded.

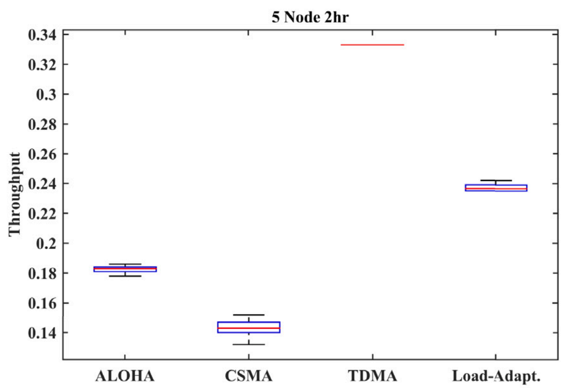
However, various limitations and challenges can be addressed for the proposed network. As mentioned, this work has considered only a scaled version of the actual topology comprising multiple ASVs and AUVs for building the MAC protocol. The protocol development of such a network should address dynamic clustering similar to that of mobile networks (cellular networks), which are implemented in multi-hop networks to form periodic dynamic clusters.

The coordination and communication among the ASVs have to be further developed. Here, it is assumed that the ASVs are in the quasi-static state. However, to build the actual topology, coordinated motion of the ASVs is required to move from the already explored region to the next unexplored region. The coordinated motion of ASVs requires proper path-planning techniques.

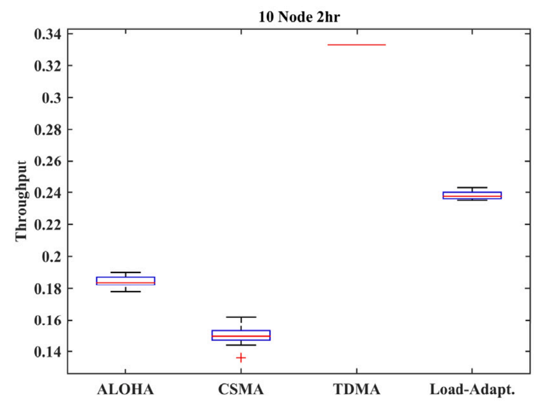
Although this work has shown the provisions of message exchanges for ranging/navigation, a proper localization algorithm can be implemented for the localization of the network as a whole (network of ASVs and AUVs) and to navigate to the frontiers successfully to map the RoI.

7. Conclusion

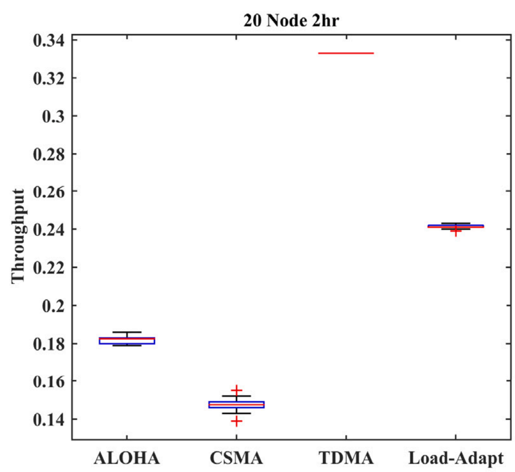
This paper designed a tailor-made MAC protocol for frontier detection to build a map of an event of interest. The topology of the multi-vehicular network comprises a hybrid fleet of vehicles. ASVs with navigational capabilities are used as leaders, and the AUVs with data collection abilities serve as followers. The ASV will lead the AUVs to



(a) Range of throughput for 5-Nodes in 2hr simulation

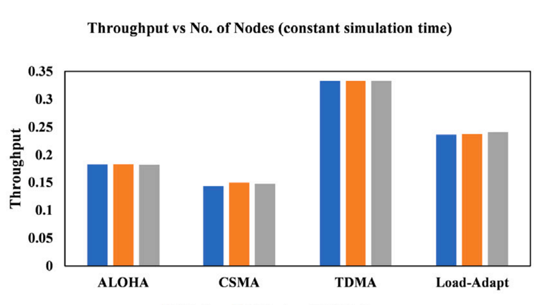


(b) Range of throughput for 10-Nodes in 2hr simulation

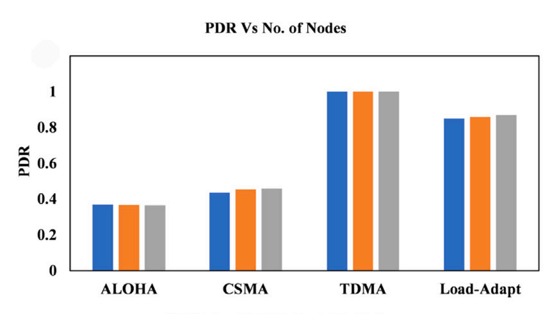


(c) Range of throughput for 20-Nodes in 2hr simulation

Fig. 11. Box plots showing the range of throughput of various protocols ALOHA, CSMA, TDMA, and Load-Adapt. MAC protocols for varying no. of nodes and constant simulation time.



(a) Throughput of various protocols with respect to no. of nodes



(b) PDR of various protocols with respect to no. of nodes

Fig. 12. Throughput vs no. of nodes and PDR vs no. of nodes for the protocols.

```
2880500|INFO|ASVNode/N@84:doInvoke|Base Node: NBTx: 79, TDMATx: 79, NBRx: 199, TDMARx:
     196, CSMARx: 124
2880500|INFO|ASVNode/N@84:call|Starting Neighbour Discovery...
2881491|INFO|AUVNode/N3@106:doInvoke|Data Node: NBRx: 73, TDMARx: 79, NBTx: 73, TDMATx
    : 61, CSMATx: 50
2881491|INFO|AUVNode/N3@106:doInvoke|Discovery request from 1, will respond after 4384
     ms
2881535|INFO|AUVNode/N1@92:doInvoke|Data Node: NBRx: 78, TDMARx: 76, NBTx: 78, TDMATx:
     69, CSMATx: 49
2881535|INFO|AUVNode/N1@92:doInvoke|Discovery request from 1, will respond after 11276
     ms
2881731|INFO|AUVNode/N2@98:doInvoke|Data Node: NBRx: 76, TDMARx: 78, NBTx: 76, TDMATx:
     68, CSMATx: 58
2881731|INFO|AUVNode/N2@98:doInvoke|Discovery request from 1, will respond after 1717
    ms
2883448|INFO|AUVNode/N2@98:call|Responding...
2884679|INFO|ASVNode/N@84:call|Node acknowledged
2885875|INFO|AUVNode/N3@106:call|Responding...
2886866|INFO|ASVNode/N@84:call|Node acknowledged
2892811|INFO|AUVNode/N1@92:call|Responding...
2893846|INFO|ASVNode/N@84:call|Node acknowledged
2900500|INFO|ASVNode/N@84:call|Broadcasting TDMA time slots...
2901491|INFO|AUVNode/N3@106:call|TDMA Started with assigned slot 2
2901535|INFO|AUVNode/N1@92:call|TDMA Started with assigned slot 0
2901535|INFO|AUVNode/N1@92:call|Transmitting in TDMA...
2901731|INFO|AUVNode/N2@98:call|TDMA Started with assigned slot 1
2902570|INFO|ASVNode/N@84:call|TDMA Data received, Location: 501,-52,-1100
2904231|INFO|AUVNode/N2@98:call|Transmitting in TDMA...
2905462|INFO|ASVNode/N@84:call|TDMA Data received, Location: 670,-453,-1100
2906491|INFO|AUVNode/N3@106:call|Transmitting in TDMA...
2907482|INFO|ASVNode/N@84:call|TDMA Data received, Location: -399,174,-1100
2908000|INFO|ASVNode/N@84:call|Starting CSMA...
2912982|INFO|AUVNode/N3@106:call|Transmitting in CSMA...
2913973|INFO|ASVNode/N@84:call|CSMA Data received, Data: [0]
2914669|INFO|AUVNode/N2@98:call|Transmitting in CSMA...
2915900|INFO|ASVNode/N@84:call|CSMA Data received, Data: [1]
```

Fig. 13. Example Log File for Load-Adaptive MAC for 4-Node scenario.

the next frontier points for the collected data from the AUVs. A load-adaptive MAC protocol is proposed here that can adjust to dynamic mobility and load in the network. The protocol ensures the reliability of the TDMA and takes advantage of the CSMA's flexibility traits. Results are generated by comparing protocols like CSMA, ALOHA, TDMA and the proposed Load-Adaptive MAC. The protocols have been compared with respect to the throughput vs number of nodes and throughput vs simulation time. It has been observed that the proposed MAC can perform better than the ALOHA and CSMA protocols. Nevertheless, it can produce comparable results for TDMA protocol while supporting the dynamic mobility and load in the network while supporting urgent data transmission for the nodes in demand.

CRediT authorship contribution statement

Ansa Shermin S.: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Formal analysis, Conceptualization. **Bhavya Mehta:** Visualization, Validation, Software, Conceptualization. **Sarang C. Dhongdi:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Mandar A. Chitre:** Visualization, Validation, Software, Resources, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgement

Sarang Dhongdi acknowledges the funding support received under the Deep Ocean Mission scheme of the Ministry of Earth Sciences, Government of India.

Appendix

An example log file from a 4-node scenario.

A complete log file of one cycle, i.e., including the ASV and AUVs communication in a basic 4-node scenario, is described in Fig. 13. The log file captures interactions between nodes in the network simulation. It includes timestamps, node interactions, network operations, node acknowledgements, and more during one protocol cycle. The operations start from the neighbour discovery, initiated by the ASV node, followed by all the mentioned operations and can be seen in the log file.

References

[1] S.S. Ansa, S.C. Dhongdi, Review of underwater mobile sensor network for ocean phenomena monitoring, J. Netw. Comput. Appl. 205 (2022) 103418.

[2] Y. Hu, H. Xie, S.-W. Jeon, H. Jin, An adaptive slotted-contention-based MAC protocol for ad-hoc networks, in: 2023 IEEE 20th Consumer Communications and Networking Conference, CCNC, 2023, pp. 648–651, http://dx.doi.org/10.1109/CCNC51644.2023.10060507.

[3] A. Rogers, K. Eshaghi, G. Nejat, B. Benhabib, Occupancy grid mapping via resource-constrained robotic swarms: A collaborative exploration strategy, Robotics 12 (3) (2023) 70.

[4] J. Hwang, N. Bose, S. Fan, AUV adaptive sampling methods: A review, Appl. Sci. 9 (15) (2019) 3145.

[5] D.O. Popa, A.C. Sanderson, R.J. Komerska, S.S. Mupparapu, D.R. Blidberg, S.G. Chappel, Adaptive sampling algorithms for multiple autonomous underwater vehicles, in: 2004 IEEE/OES Autonomous Underwater Vehicles (IEEE Cat. No. 04CH37578), IEEE, 2004, pp. 108–118.

[6] W. Burgard, M. Moors, C. Stachniss, F.E. Schneider, Coordinated multi-robot exploration, IEEE Trans. Robot. 21 (3) (2005) 376–386.

[7] G.A. Hollinger, G.S. Sukhatme, Sampling-based robotic information gathering algorithms, Int. J. Robot. Res. 33 (9) (2014) 1271–1287.

[8] G. Ferri*, A. Munafò*, A. Tesei, P. Braca, F. Meyer, K. Pelekanakis, R. Petroccia, J. Alves, C. Strode, K. LePage, Cooperative robotic networks for underwater surveillance: an overview, IET Radar Sonar Navigat. 11 (12) (2017) 1740–1761.

[9] R.B. Wynn, V.A. Huvenne, T.P. Le Bas, B.J. Murton, D.P. Connelly, B.J. Bett, H.A. Ruhl, K.J. Morris, J. Peakall, D.R. Parsons, et al., Autonomous underwater vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience, Marine Geol. 352 (2014) 451–468.

[10] F. Guerra, P. Casari, A. Berni, J. Potter, M. Zorzi, Performance Evaluation of Random and Handshake-based Channel Access in Collaborative Mobile Underwater Networks, in: OCEANS 2010 MTS/IEEE SEATTLE, IEEE, 2010, pp. 1–7.

[11] A. Caiti, V. Calabrò, A. Munafò, AUV team cooperation: Emerging behaviours and networking modalities, IFAC Proc. Vol. 45 (27) (2012) 342–347.

[12] N.-Y. Yun, Y.-P. Kim, S. Muminov, J.-Y. Lee, S.-Y. Shin, S.-H. Park, Sync MAC protocol to control underwater vehicle based on underwater acoustic communication, in: 2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing, IEEE, 2011, pp. 452–456.

[13] M. Deng, H. Chen, L. Xie, DCO-mac: A hybrid MAC protocol for data collection in underwater acoustic sensor networks, Sensors 18 (7) (2018) 2300.

[14] Z. Zhang, W. Shi, Q. Niu, Y. Guo, J. Wang, H. Luo, A load-based hybrid MAC protocol for underwater wireless sensor networks, IEEE Access 7 (2019) 104542–104552.

[15] R. Diamant, L. Lampe, A Hybrid Spatial Reuse MAC Protocol for Ad-hoc Underwater Acoustic Communication Networks, in: 2010 IEEE International Conference on Communications Workshops, IEEE, 2010, pp. 1–5.

[16] M. Molins, M. Stojanovic, Slotted FAMA: a MAC Protocol for Underwater Acoustic Networks, in: OCEANS 2006-Asia Pacific, IEEE, 2006, pp. 1–7.

[17] C. Petrioli, R. Petroccia, M. Stojanovic, A comparative performance evaluation of MAC protocols for underwater sensor networks, in: OCEANS 2008, IEEE, 2008, pp. 1–10.

[18] L. Kleinrock, F. Tobagi, Packet Switching in Radio Channels: Part I-Carrier Sense Multiple-Access Modes and their Throughput-Delay Characteristics, IEEE Trans. Commun. 23 (12) (1975) 1400–1416.

[19] P. Karn, et al., MACA-a new channel access method for packet radio, in: ARRL/CRRL Amateur Radio 9th Computer Networking Conference, Vol. 140, London, Canada, 1990, pp. 134–140.

[20] C.L. Fullmer, J. Garcia-Luna-Aceves, Floor acquisition multiple access (FAMA) for packet-radio networks, in: Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, 1995, pp. 262–273.

[21] B. Peleato, M. Stojanovic, Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks, IEEE Commun. Lett. 11 (12) (2007) 1025–1027.

[22] Y. Zhang, H. Chen, W. Xu, A load-adaptive CSMA/CA MAC protocol for mobile underwater acoustic sensor networks, in: 2018 10th International Conference on Wireless Communications and Signal Processing, WCSP, IEEE, 2018, pp. 1–7.

[23] M. Stojanovic, L. Freitag, J. Leonard, P. Newman, A network protocol for multiple AUV localization, in: OCEANS'02 MTS/IEEE, Vol. 1, IEEE, 2002, pp. 604–611.

[24] X. Liu, X. Du, M. Li, L. Wang, C. Li, A MAC protocol of concurrent scheduling based on spatial-temporal uncertainty for underwater sensor networks, J. Sensors 2021 (2021) 1–15.

[25] D. Centelles, A. Soriano, J.V. Marti, P.J. Sanz, Underwater multirobot cooperative intervention MAC protocol, IEEE Access 8 (2020) 60867–60876.

[26] A.-R. Cho, C. Yun, Y.-K. Lim, Y. Choi, Asymmetric propagation delay-aware TDMA MAC protocol for mobile underwater acoustic sensor networks, Appl. Sci. 8 (6) (2018) 962.

[27] S. Shahabudeen, M. Chitre, M. Motani, Adaptive multimode medium access control for underwater acoustic networks, IEEE J. Ocean. Eng. 39 (3) (2013) 500–514.

[28] F. Salva-Garau, M. Stojanovic, Multi-cluster protocol for ad-hoc mobile underwater acoustic networks, in: Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492), Vol. 1, IEEE, 2003, pp. 91–98.

[29] J. Śliwka, R. Petroccia, A. Munafò, V. Djapic, Experimental evaluation of net-LBL: An acoustic network-based navigation system, in: OCEANS 2017-Aberdeen, IEEE, 2017, pp. 1–9.

[30] B.P. Crosbie, M. Anderson, E. Wolbrecht, J. Canning, D.B. Edwards, Synchronous navigation of AUVs using WHOI micro-modem 13-bit communications, in: OCEANS 2010 MTS/IEEE SEATTLE, 2010, pp. 1–5.

[31] M. Bresciani, S. Tani, L. Bazzarello, A. Caiti, R. Costanzi, Acoustic positioning of multiple AUVs by an ASV: experimental validation and performance characterisation, IFAC-PapersOnLine 56 (2) (2023) 5753–5758.

[32] A.Y. Rodionov, P. Unru, A.Y. Kulik, F. Dubrovin, Model of adaptive navigation and communication protocol for the organization of AUV group operation, in: 2017 IEEE Underwater Technology, UT, IEEE, 2017, pp. 1–7.

[33] G. Ferri, A. Munafò, J. Alves, K. LePage, A data-driven control strategy in synergy with continuous active sonar for littoral underwater surveillance, in: OCEANS 2016 MTS/IEEE Monterey, IEEE, 2016, pp. 1–7.

[34] M. Carreras, J. Batlle, P. Ridao, G. Roberts, An overview on behaviour-based methods for AUV control, IFAC Proc. Vol. 33 (21) (2000) 141–146.

[35] C.R. German, D.R. Yoerger, M. Jakuba, T.M. Shank, C.H. Langmuir, K.-i. Nakamura, Hydrothermal exploration with the autonomous benthic explorer, Deep Sea Res. I: Oceanogr. Res. Pap. 55 (2) (2008) 203–219.

[36] W. Li, J.A. Farrell, S. Pang, R.M. Arrieta, Moth-inspired chemical plume tracing on an autonomous underwater vehicle, IEEE Trans. Robot. 22 (2) (2006) 292–307.

[37] T. Balch, R.C. Arkin, Behavior-based formation control for multirobot teams, IEEE Trans. Robot. Autom. 14 (6) (1998) 926–939.

[38] D. Fenucci, J. Sitbon, J. Neasham, A. Phillips, A. Munafò, Ad hoc acoustic network aided localization for micro-AUVs, Field Robot. 2 (1) (2022) 1888–1919.

[39] Y. Xu, M. Li, S. Chen, Z. Tao, P. Pei, W. Feng, Self-adaptive MAC and routing protocols for mobile underwater acoustic sensor networks, in: OCEANS 2021: San Diego – Porto, 2021, pp. 1–9, http://dx.doi.org/10.23919/OCEANS44145.2021.9706055.

[40] A. Munafò, J. Sliwka, G. Ferri, A. Vermeij, R. Goldhahn, K. LePage, J. Alves, J. Potter, Enhancing AUV localization using underwater acoustic sensor networks: Results in long baseline navigation from the COLLAB13 sea trial, in: 2014 Oceans-St. John's, IEEE, 2014, pp. 1–7.

[41] M. Chitre, R. Bhatnagar, W.-S. Soh, UnetStack: An agent-based software stack and simulator for underwater networks, in: 2014 Oceans-St. John's, IEEE, 2014, pp. 1–10.

Ansa Shermin S is a Ph.D. scholar in the Dept. of EEE at BITS Pilani K K Birla Goa Campus, Goa, India. She completed her B.Tech. in Electronics and Communication Engineering from Kerala University in 2012 and M.Tech in Communication Systems from Kerala University in 2015. Her research interests include Mobile Underwater Acoustic Communication and Sensor Networks, Wireless Sensor Networks and Swarm Networks.

Bhavya Mehta is a final year student at BITS Pilani KK Birla Goa Campus, Goa, India, pursuing a dual degree in M.Sc in Mathematics and B.E. in Electronics and Communications Engineering. His academic journey reflects a keen interest in fields of Robotics, Digital Image Processing, Communication Networks, and Digital Signal Processing.

Dr Sarang C. Dhongdi is Assistant Professor in the Dept. of EEE at BITS Pilani K K Birla Goa Campus, Goa, India. He completed his Ph.D. in the area of Underwater Acoustic Sensor Networks in the year 2017. His research interests include Underwater Acoustic Communication and Sensor Networks, Wireless Sensor Networks, Flying ad-hoc networks and Digital Signal Processing.

Mandar Chitre is currently the Head of the ARL at Tropical Marine Science Institute (TMSI) and an Associate Professor at the Department of Electrical & Computer Engineering (ECE) of the National University of Singapore (NUS). Before joining the ARL in 2003, he headed the technology division for a Singapore-based telecommunication software company for 5 years. Mandar was awarded a Ph.D. by NUS for his research in underwater acoustic communications in warm, shallow waters. Before this, he obtained a Master's degree from NUS, developing simulations and algorithms for Ambient Noise Imaging, and another in Bioinformatics from the Nanyang Technological University (NTU). Mandar's research interests include underwater acoustic signal processing, underwater communications & networking, ambient noise imaging (ANI), and cooperative underwater robotics. Mandar served as the Editor-in-Chief for the IEEE Journal of Oceanic Engineering from 2018 to 2023. He has also served on several editorial boards and conference committees.