Wireless Personal Communications (2024) 134:781–808
https://doi.org/10.1007/s11277-024-10926-6

Cluster-Based Multi-attribute Routing Protocol for Underwater Acoustic Sensor Networks

Pradeep Nazareth[1] · **B. R. Chandavarkar**[1]

Accepted: 27 February 2024 / Published online: 28 March 2024

Abstract
Underwater Acoustic Sensor Networks play a significant role in various underwater applications. There are several challenges in underwater communications like high bit-error-rate, low bandwidth, high energy consumption, void-node during routing, etc. Handling void-node during routing is a major challenge in underwater routing. There are well-known void-handling protocols like Energy-efficient Void-Aware Geographic Routing protocol, HydroCast, etc. However, these routing protocols require all neighboring nodes must be a part of the cluster which increases the overhead on clustering, or void-node has a part of the routing. This paper proposes an underwater routing protocol referred to as Cluster-based Multi-Attribute Routing (CMAR) to overcome these issues. It is a sender-based, opportunistic underwater routing protocol. CMAR uses the Technique for Order of Preference by Similarity to Ideal Solution to evaluate the suitability of the neighboring nodes and the basis for clustering process initialization. Through MATLAB simulations, the performance of the CMAR is compared with HydroCast in terms of the number of nodes selected in the forwarding set, number of clusters formed, number of times void-node becomes part of routing and transmission reliability.

Keywords Underwater routing · Clustering · MADM · TOPSIS

1 Introduction

Water covers the majority portion of the surface of the earth. Exploring underwater is a growing field of research in communication. Underwater Acoustic Sensor Networks (UASNs) are the technology that enables exploring of underwater. UASNs have several applications such as underwater environment monitoring, underwater resource exploration, detection of various underwater disturbances, underwater surveillance, the study of marine

✉ Pradeep Nazareth
pradeep.nitk2017@gmail.com

B. R. Chandavarkar
brcnitk@gmail.com

[1] Department of Computer Science and Engineering, Wireless Information Networking Group (WiNG), National Institute of Technology Karnataka, Srinivasnagar, Surathkal, Mangalore, Karnataka 575 025, India

life, etc. [1]. On the other end, UASNs have challenges such as high bit-error-rate (BER), low bandwidth, node movement, energy constraint, varying link quality between nodes, and void-node during routing [2–5]. These challenges have a significant impact on routing.

In the literature number of wireless routing protocols exist. They are mainly classified into proactive, reactive, and geographic routing protocols as shown in Fig. 1. The proactive routing protocols aim at reducing end-to-end delay in delivering packets. It maintains the up-to-date path from each node to all other nodes in the network. However, proactive routing incurs overhead by transmitting control packets while establishing the path for the first time and subsequently during changes in network topology. Updated information must be propagated throughout the network when the topology is changed. The approach can determine the path from one node to all other nodes in the topology. It is not required in the case of UASNs at the cost of transmission of multiple control packets, which consume the number of resources of underwater nodes. However, in reactive routing, the path to the sink (destination) is discovered when packets need to be delivered to the sink. Thus it results in high end-to-end delay in delivering packets, which is not suitable for UASNs [4–6]. The geographic routing protocols overcome the issues of the proactive and reactive routing protocols for underwater communication.

Geographic routing uses greedy forwarding. In the greedy forwarding technique, the forwarding node finds only the next-hop node instead of finding the complete path from the source to the sink. In greedy forwarding, a node forwards packets to a neighboring node closest to the sink. The forwarding node is referred to as void-node if it is failed to find the neighbor with positive progress [7, 8]. The lack of void-node handling capability of the routing protocol creates unreliable forwarding, high end-to-end delay, looping during routing, high-energy consumption, etc. [8, 9]. UASNs is deployed in a different environment than the terrestrial networks; thus, different void-handling techniques are used for underwater networks. Some of the challenges of the void-handling in the UASNs are node mobility, void caused by the environment, and the three-dimensional nature of the topology [8, 10]. Therefore, designing an appropriate void-handling strategy is essential to mitigate the void nodes' impact.

Geographic routing protocols can be either opportunistic or non-opportunistic, as shown in Fig. 1. In non-opportunistic routing, a forwarding node selects a single next-hop to forward the received packets. There are high chances of failure of packet transmissions. In the Opportunistic Routing (OR) approach, a forwarding node selects a

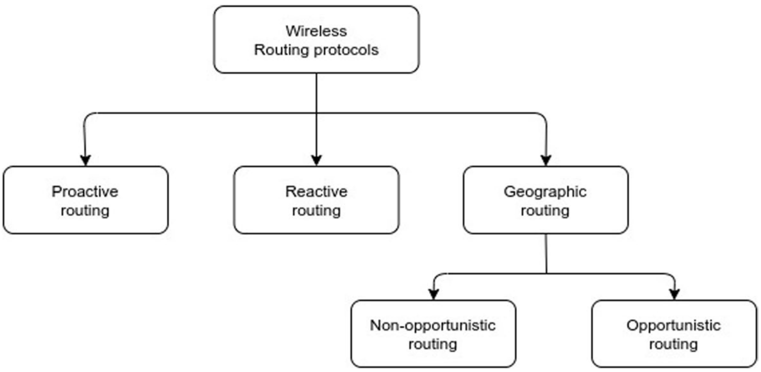

Fig. 1 Classification of wireless routing protocol

potential set of nodes referred to as a candidate forwarding set. Nodes in the candidate forwarding set that receives the transmitted packet and coordinate among themself to select the forwarding node. The advantage of OR is it improves the performance of routing [11–13].

Some of the state-of-the-art void-handling opportunistic underwater routing protocols are HydroCast [14], Energy-efficient Void-Avoidance Geographic Routing (EVAGR) [9]. These routing protocols use multiple attributes such as progress and packet delivery probability with their neighbors to decide their candidate forwarding set. Normalized Advancement (NADV) of all neighbors present at lower depth is calculated. Further, the candidate forwarding set is determined based on forming clusters. All the neighbors present at lower-depth must be a part of one of the clusters. Additionally, by computing Expected Packet Advancement (EPA), the best cluster is selected, and nodes of the selected cluster form a candidate forwarding set. The significance of the clustering is to overcome the duplicate packet transmission and hidden-node terminal issue. One of the disadvantages of this approach is that even though data is forwarded to one cluster, they are required to form many clusters. It increases the complexity of the clustering [9, 14, 15]. Further, to achieve coordination among nodes in the selected cluster, they must compute the hold time. Hold time defines the waiting period of a node before forwarding the received packet. HydroCast and EVAGR require distance among nodes in the selected cluster to calculate hold time. It introduces additional overhead on the nodes.

In the proposed Cluster-based Multi-Attribute Routing (CMAR) protocol, weights for individual attributes are computed using entropy weight method [16]. Further, nodes at lower-depth and non-void neighboring nodes are evaluated by applying the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). TOPSIS is a well-known Multi-Attribute Decision Making (MADM) method used in decision-making in various applications [17]. The TOPSIS method considers given input attributes of neighbor nodes and generates the scores. The score indicates the preference of the node among neighbors. The set of attributes that can be considered to evaluate the neighboring nodes are:

- *Hop-count:* Underwater routing protocols like LLSR [18], LFVAR [19], IVAR [20], OVAR [15] uses hop-count as one of the metrics to select the next-hop or candidate forwarding set. Lower the hop-count, better the node to be selected as next-hop.
- *Path quality:* It is the count of the redundant paths to reach the sink node. The LLSR protocol [18] uses the path quality of neighboring nodes to select the next-hop. The higher path quality of the neighboring node indicates that it has more paths to reach the sink.
- *Link quality:* The link quality affects the performance of underwater routing. Link quality is measured in received signal strength and error rate. The CARP [21] uses link quality as one of the parameters to select next-hop. Lower the error rate or higher the received signal strength results in improved link quality.
- *Residual energy of neighbor:* Energy is the major constraint in the underwater network. Nodes drain their energy during trans-receiving activity. The residual energy of the neighboring node plays an essential role in selecting the next-hop, as more residual energy is preferred. The energy consumed to transmit n bits of data to a distance of d is given in the Eq. (1) [22].

$$E_{tx}(n,d) = n * E_{ele} + n * T_b * C * H * d * e^{\alpha(f)*d} \tag{1}$$

where E_{ele} is the energy consumed by the transmitter circuit, T_b is the bit duration while transmitting the data, H is the water depth, $\alpha(f)$ is the absorption coefficient computed as per Eq. (33), C is the empirical constant given by the equation,

$$C \approx 2\pi * 0.67 * 10^{-9.5} \tag{2}$$

To receive n bit packet in underwater, energy consumption is given by the equation,

$$E_{rx}(n,d) = n * E_{ele} \tag{3}$$

Apart from the above attributes, further attributes like packet delivery probability the progress of the neighboring node with respect to the sender can be considered. In TOPSIS, attributes of neighboring nodes are either benefit or cost attributes. The benefit attributes, where the higher the value, is preferred. For example, residual energy, packet delivery probability, node progress, path quality, link quality, etc., are considered benefit attributes. In the case of cost attribute, the higher the attribute value, the lower the preference for next-hop selection. Some of the examples are queuing delay, hop-count, etc. CMAR considers void status as neither a benefit attribute nor a cost attribute. The void status of the neighbor is used to filter out the neighbors.

The CMAR protocol uses a new clustering technique in which there is no need to cluster all neighboring nodes. CMAR allows to specification of the Threshold Number (TN) of nodes that must be clustered, and the clustering process continues until the TN of nodes is clustered. Once clustering is completed, nodes in the cluster(s) are part of the forwarding set. Further, achieving coordination among nodes in the selected cluster requires hold time computation in opportunistic routing. Thus CMAR introduced a new hold time function based on the priority of nodes. It doesn't require obtaining the distance among the nodes in the cluster.

Motivation: For underwater routing, different routing protocols are proposed. To provide improved performance, there are many opportunistic routing protocols proposed. Most routing protocols use multiple attributes to decide on forwarding set or next-hops. However, no protocols consider determining appropriate weights for the selected attributes and obtaining an overall score using TOPSIS. Further, existing opportunistic routing protocols use clustering techniques, which need clustering of all neighbor nodes, increasing overhead. The proposed CMAR underwater protocol overcomes these deficiencies. The proposed CMAR determines weights to the attributes by applying an entropy weight method, and the TOPSIS is used to find the suitability of neighbor nodes. The CMAR simplifies the clustering approach by clustering only the TN of nodes. It results in a reduced number of clusters and reduces the complexity of clustering. The significant contributions of the paper are highlighted as follows:

- This paper designed a novel underwater routing protocol referred to as Cluster-based Multi-Attribute Routing (CMAR) protocol.
- The candidate forwarding set is determined by forming cluster(s). It avoids the need for clustering of all neighbors. It includes the specified Threshold Number (TN) of nodes in the cluster(s). Thereby, it reduces the complexity of the clustering.

- CMAR avoids void nodes as part of the cluster(s).

The organization of the paper is as follows: Sect. 2 elaborates on the existing underwater void-aware routing protocol that exists in the literature. Section 3 describes the detailed design of the CMAR protocol, and Sect. 4 discusses the implementation of the CMAR protocol. Detailed results and their analysis are carried out in Sect. 5. The paper is concluded with the conclusion in Sect. 6.

2 Related Work

There are several underwater routing protocols in the literature. Most of the routing protocols make routing decisions which are relied on multiple attributes of the neighboring nodes. Routing protocols are classified as void-ignorance or void-handling based on their ability to handle void nodes. The void-ignorance routing protocols do not have any mechanism to address void-node issues. Void-handling routing protocols have mechanisms to handle void nodes either by determining recovery paths through void nodes or avoiding void nodes during routing. As illustrated in Fig. 2, void-handling routing protocols are divided into location-based and depth-based underwater routing protocols.

- Location-based routing protocols: All nodes should be aware of their three-dimensional coordinate information. It is achieved by using underwater localization algorithms [23].
- Depth-based routing protocols: These types of routing methods rely solely on node depth information to route packets. A pressure gauge is placed on each node to determine its depth underwater [6].

2.1 Location-Based Routing Protocols

In this section, location-based void-handling routing protocols are elaborated with reference to their routing criteria and mechanisms used to deal with void nodes in the UASNs. The Vector-Based Void Avoidance (VBVA) underwater routing protocol, proposed by Xie Peng et al. [10], is a vector-based underwater routing protocol. It is a receiver-based, opportunistic routing protocol. In the absence of the void nodes, the protocol operation of VBVA is identical to VBF [24]. When a node receives the packet, it finds its desirableness

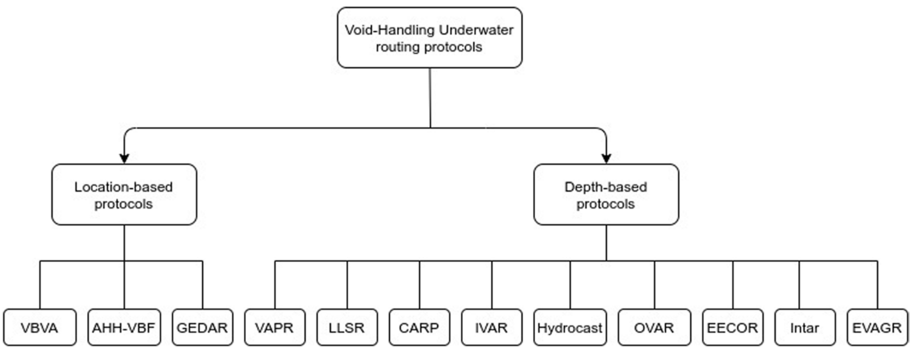

Fig. 2 Classification of void-handling protocols for underwater

factor. The desirableness factor is calculated based on the distance between the sender and the node, the width of the virtual pipe surrounding the vector, the node's projection to the routing vector between the source and the sink, the transmission range, and angle between sender and the source. A lower desirableness factor of the node indicates the more preferred node to forward the packet and has a lower holding time. VBVA uses vector-shift to handle the convex void issue [8]. When a sender fails to overhear the forwarding of the data packet by its neighbor, it broadcasts a control packet referred to as vector-shift. The vector shift asks neighbors to change the forwarding vector and is involved in the forwarding of packets. The node will broadcast a back-pressure (BP) control packet when a packet is stuck at a concave void. Upon receiving the BP packet, neighboring nodes shift forwarding vectors for that packet. Still, the node does not find any eligible nodes, the process of broadcasting the BP packet continues until a node finds an eligible forwarding node [8, 10]. However, the void-recovery mechanism used in VBVA is too complicated, and recovery from the concave node is time-consuming. VBVA does not prevent packets from trapping at the concave void [8].

The issue of packets trapping at concave void-node in the VBVA is eliminated by the Adaptive Hop-by-Hop Vector-Based Forwarding (AHH-VBF) [25]. It is a preventive void avoidance routing protocol. Each node in the network is aware of three-dimensional information of its own, neighbors, sink, and sender of the packet. To handle the void-node, it uses two approaches: changing the virtual pipe's direction and adjusting the radius of the virtual pipeline between the forwarding node and the sink node. It helps find more candidate nodes in the sparse region, improving transmission reliability. AHH-VBF has a transmission power adjustment feature. Each node knows the neighbor's location, which helps select the proper power level. In sparse regions, transmission power is increased to cover more nodes, and in dense areas, it reduces transmission power to reduce energy consumption. Even though AHH-VBF can adjust the radius of virtual pipe and transmission power, it can handle only small void regions. Thus it will not always guarantee packet delivery. It is not suitable in an environment where network topology is rapidly changing. To ensure the guaranteed packet delivery and to handle the void nodes, Geographic and opportunistic routing with Depth Adjustment-based topology control for communication Recovery (GEDAR) is proposed [26]. It is a geographic and opportunistic routing protocol. GEDAR uses complete location information of neighbors. The GEDAR protocol's novelty is identifying all void nodes in the network. Further, it uses depth adjustments to transform the void nodes to normal nodes compared to traditional message-based void recovery mechanisms.

2.2 Depth-Based Routing Protocols

In this routing protocol category, routing takes place using the depth information of neighboring nodes without knowing full geographical coordinates. It eliminates the need for costly localization; thereby, it reduces the energy consumption of the nodes. This subsection discusses the depth-based, void-handling underwater routing protocols.

Void-Aware Pressure Routing (VAPR) is an opportunistic protocol [27]. A preventive void-handling mechanism is used by routing the data packets away from void nodes during routing. Its operation consists of two phases—enhanced beaconing and opportunistic directional forwarding. The enhanced beaconing involves developing direction trails. The beacons are periodically generated from the sink nodes comprising attributes like hop-count, packet sequence number, forwarding-direction, and depth of the node. The sequence number and the hop-count information set the node's data forwarding direction. Further, by comparing the direction (up or down) from which it receives the beacon and by extracting the sender's data forwarding direction, it can differentiate normal and void and trap neighbor. Thereby avoiding void and trap nodes. However, VAPR avoids the void nodes by holding information up to two hops, resulting in high overhead in the system.

However, the issue of high overhead is overcome by the Location-free Link State Routing (LLSR) [18], which is a void-avoidance routing protocol. A node chooses a next-hop based on attributes like hop-count, path quality, and pressure attributes of neighbors. Initially, a sender or forwarding node considers the neighbor's minimum hop-count as its next-hop. In the case of a tie, the sender considers the path quality of the nodes among those neighbors and selects a neighbor having the highest path quality. If the tie persists, the node chooses a neighbor at the lowest depth as the next-hop. The attributes are distributed across the nodes in the network through the periodic beacons. The beacons are originated from the sink and subsequently propagated by the other nodes in the network. However, only one node is selected as the next-hop, which may result in a lower successful packet transmission rate [8]. Further, the hop-count value depends on the beacon period. The hop-count value may not reflect the up-to-date network topology if the beacon period is too large. If the beacon period is too small, the network is loaded with beacons and consumes more energy. LLSR will not consider the link quality between the sender and the neighboring node when the next-hop is selected. There is a possibility of poor link quality between the sender and the selected next-hop.

Stefano Basagni et al. [21] proposed Channel-Aware Routing Protocol (CARP) addresses issues of LLSR by selecting next-hop based on goodness and hop-count. Goodness is the product of the two-hop link quality of the sender. The link quality between nodes is measured using Exponentially Weighted Moving Average (EWMA). The hop-count in CARP is more sensitive to the link quality than the actual hop distance from the sink. The hop-count is computed by maintaining a variable referred to as L_{ratio_i}, where i is the hop-count. For every hop-count, i maintains the ratio of the number of packets acknowledged to the total number of packets sent by the sender. Further, the sender will consider the minimum hop-count such that L_{ratio_i} is above the threshold values. However, CARP is the non-opportunistic routing protocol that chooses only a neighbor as its next-hop.

Inherently Void-Avoidance Routing (IVAR) is a depth-based, opportunistic underwater routing protocol [20]. It has two phases, periodic beaconing, and routing. The main aim of periodic beaconing is to generate hop-count awareness among nodes in the network. The sink node triggers the beacons and progresses towards the leaf nodes. The beacons are generated periodically to reflect the changes in the topology. In the routing phase, the forwarding node broadcasts the data packet. The data packet consists of data, depth of forwarding node, sender hop-count, source ID, and packet sequence number. When neighboring nodes receive the data packet, they check whether it is a duplicate or a new packet. If it is a duplicate packet, the neighboring node discards the packet. If the hop-count of neighboring nodes is smaller than that of the sender, then those neighboring nodes are considered candidate forwarding nodes to further forward the data packet. Further, all candidate nodes compute their hold time. A candidate node having the lowest hold time will first forward the data packet. Other nodes overhear the transmission are suppress the transmission of the same data packet. Hold time is the function of hop-count, depth difference between the sender and the receiver, and relative distance between the sender node and the candidate node. However, even though IVAR improves transmission reliability by using an opportunistic routing technique, there are chances of duplicate packet transmission and hidden-node problems in case of candidate nodes are out of communication range of each other [8].

Youngtae noh et al. [14] proposed HydroCast underwater routing protocol, which eliminates duplicate packet and hidden-node terminal issues of IVAR. It makes routing decisions based on local attributes such as progress and packet delivery probability. Normalized Advancement (NADV) of all neighbors is computed. Using NADV & neighborhood information of node, clusters are formed. Further, using cluster information and node priority within the cluster, Expected Packet Advancement (EPA) will be computed. A cluster with maximum EPA is selected as the best cluster. The sender will forward the packet to the chosen cluster opportunistically. HydroCast determines recovery paths to bypass void areas during data forwarding. Thus, it leads to an increased end-to-end delay in encountering void-node in the network.

Opportunistic Void Avoidance Routing (OVAR) selects neighbors with a lower hop-count than the sender to avoid void-nodes [15]. It uses one-hop neighbor information to be aware of neighbor node information. The forwarding set is selected by computing Expected Packet Advancement (EPA). The number of nodes in the forwarding set is adjusted to balance the trade-off between reliability and energy. Energy-efficient cooperative opportunistic routing (EECOR) [28] is a single sink node routing protocol. Source node obtains depth and energy information of the neighboring nodes. In EECOR, initially forwarding set is established. Further, the best relay set is selected & forwarded packet to these nodes by using fuzzy logic-based relay selection. By using coordination among selected best node relay nodes, high priority node forwards the packets, and other nodes in the best relay set on overhearing transmission suppress the transmission of the same packet.

Interference-Aware Routing (Intar) is a depth-based underwater routing [29]. Nodes in the networks determine all the neighbors at lower depth are referred to as potential Forwarding Nodes(PFN). Further, the sender node computes the cost function of all the PFNs and selects a PFN having the highest cost function value. The cost function depends on the distance between sender and PFN, hop-count, and the number of neighbors of PFN. Energy-efficient void avoidance geographic routing protocol for underwater sensor networks (EVAGR) [9] is selecting its candidate set by computing Expected Packet Advancement (EPA) of neighbors in positive advancement. Further, by eliminating void nodes from the candidate set, the weight of individual nodes in the candidate set is computed. Finally, only half of the nodes from the candidate set having the highest weight are shortlisted as forwarding set.

The Location-Free Void Avoidance Routing (LFVAR) protocol which is a sender-based protocol [19]. It picks the best non-void node during routing. The selection of the best node is done using the computation of the cost function. The cost function depends on the progress and the hop-count of the neighbor. However, the LFVAR protocol uses a non-opportunistic routing approach resulting in lower reliability of transmission.

Distance-Vector-based Opportunistic Routing (DVOR) is a receiver-based, opportunistic routing protocol [30]. The Data packet in DVOR consists of the source id, the hop-count of the previous forwarder, and data. When the data packet arrives at the nodes, only those nodes with a hop-count lower than the last forwarder are eligible to forward the data further. The hop-count information is propagated in the network by periodic beaconing. Moreover, eligible forwarding nodes compute their hold time. The hold time depends on the random number between 0 and the back-off window. The major limitations of the DVOR are duplicate packet transmission and hidden-node problems. The Reinforcement Learning-based Opportunistic Routing (RLOR) Protocol can avoid void nodes during routing [31]. Void nodes are identified by the number of neighbors above the node as a reward function. In the case of no neighbors being present above the node, the reward is considered as 0. Thereby, the probability of selecting a void node is too low. However, DVOR and RLOR protocols increase the chance of duplicate transmission and packet collision during routing as receiving nodes are not aware of each other's data forwarding.

To per best of our knowledge, no underwater routing protocols are proposed that use the Multi-Attribute decision-making (MADM) and weight for attributes approach. Thus in this paper, weights of attributes are calculated, and the MADM approach has been used to calculate the score of neighbor nodes. Further, cluster(s) are formed using the scores of neighboring nodes. Additionally, a novel hold time function is proposed to coordinate cluster nodes.

3 Cluster-Based Multi-Attribute Routing (CMAR) Protocol

This section elaborates on the design of the proposed CMAR protocol. The system model is described in Sect. 3.1. CMAR consists of two phases, and they are described as follows:

- *Phase - I: Score computation and cluster formation:* Score computation and cluster formation are the essential steps to determine the candidate forwarding set. Formation of the cluster(s) is initiated based on the scores of the Potential Relay Nodes (PRNs). The method to obtain PRNs is described in Sect. 3.1. Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) calculates the score. Steps in score computation are elaborated in Sect. 3.2.1. Further, clustering is done to select the candidate forwarding set. A novel approach has been used to form the cluster(s) in the CMAR. In the CMAR, the clustering process includes the TN of nodes in the cluster(s). Section 3.2.2 elaborates on the clustering technique used in the CMAR.
- *Phase - II: Forwarding of received data:* As nodes in the candidate forwarding set receive the data, further packet forwarding is coordinated by defining holding time. The receiving node has to keep the received packet in its buffer until the hold time expires or overhears the transmission of the same data packet by another node in the candidate forwarding set. A new holding time function is proposed and elaborated in Sect. 3.3.

3.1 System Model

This section elaborates on the system model used in the CMAR. Assume that the source/forwarder node i has a set of neighbor nodes j. Node i selects a suitable set of neighbors referred to as Potential Relay Nodes PRN_i. The PRN_i is computed as per the Eq. (4).

$$PRN_i = \{\forall j \in N_i \ \&\& \ V_j = 0 \ \&\& \ Depth(j) < Depth(i)\} \quad (4)$$

In the Eq. (4), N_i is the set of neighboring nodes of node i, V_j indicates the void-status of the node j, $V_j = 0$ indicates node j is normal node and $Vj = 1$ indicates node j is void-node and *Depth*(j) indicates the depth at which node j is deployed. The Eq. (4) indicates PRN contains only non-void (normal) nodes present at a lower depth than the source/forwarder node. The CMAR selects forwarding nodes from the PRNs of the source/forwarder node by calculating scores and applying the clustering formation algorithm. The score calculation and clustering algorithm are elaborated in the following subsections.

3.2 Phase-I: Score Calculation and Cluster Formation

In this subsection, score calculation and cluster formation method is elaborated. TOPSIS approach has been used in CMAR to calculate the scores of PRNs of the sender or forwarding node. TOPSIS is the well-known & widely used technique in the MADM [32, 33]. A set of attributes of PRNs act as the input to the TOPSIS. The sender or forwarding node computes the scores of the PRNs.

3.2.1 Score Calculation Using TOPSIS

In the CMAR, PRN_i is determined using the Eq. (4), and attributes of nodes in PRN_i act as the input to the TOPSIS. The steps involved in computing scores of nodes in PRN_i by the source/forwarder i are elaborated as follows:

Step 1: Construction of decision matrix of nodes in PRNs

This step involves the creation of a decision matrix representing n attributes (advancement and PDP) of m nodes of PRN_i. Nodes are represented in rows, and attributes of the corresponding node are represented in columns. The intersection of x^{th} node with y^{th} attribute is represented by $DM^i[x][y]$. Thus, decision matrix of nodes in PRN_i are represented as per Eq. (5).

$$(DM^i[x][y])_{m \times n} \quad (5)$$

where $DM^i[x][y]$ holds the y^{th} attribute of x^{th} node in PRN_i, $m = |PRN_i|$, and n is the number of attributes to be considered, $x \in \{1, 2, 3, ...m\}$, and $y \in \{1, 2, 3, ...n\}$

Step 2: Attribute normalization

This step involves normalization of decision matrix $((DM^i[x][y])_{m \times n})$. Normalization of attributes involves the transformation of the attributes from dimension to dimensionless and put into a common scale, so that they are comparable [34]. In CMAR, the sum normalization method is used to normalize the attributes. The Eq. (6) indicates, normalization of attribute at y^{th} column, corresponding to node at x^{th} row of the matrix given in Eq. (5).

$$R^i[x][y] = \frac{DM^i[x][y]}{\sum_{x=1}^{m} DM^i[x][y]} \quad (6)$$

The normalized matrix is represented as shown in Eq. (7).

$$(R^i[x][y])_{m \times n} \quad (7)$$

Step 3: Weight computation

Normalized matrix is obtained from the decision matrix of source/forwarder i. The attributes used in the decision matrix or normalized matrix do not have equal importance. Every attribute has to have a different priority. The weight of the attributes indicates the importance or priority. Thus, the weights of each attribute need to be determined. There are many techniques to compute the weights; some of them are Simplified and Improved Analytical Hierarchy Process (SI-AHP) [35], Entropy, Variance [36] etc. However, the Entropy weighting method is appropriate for the TOPSIS when the Sum normalization method is used [33]. Thus CMAR uses the Entropy weight method to compute the weights of the attribute. The computation of weight of the attribute at y^{th} column, $W^i[y]$ using Entropy weight method is shown in Eq. (8).

$$W^i[y] = \frac{1 - E^i[y]}{\sum_{y=1}^{n}(1 - E^i[y])} \tag{8}$$

where $E^i[y]$ is the entropy of attribute at y^{th} column, and computed using Eq. (9).

$$E^i[y] = -\frac{\sum_{x=1}^{m}(R^i[x][y] * ln(R^i[x][y])}{ln(m)} \tag{9}$$

The weight matrix is represented as shown in Eq. (10).

$$(W^i[y])_{1\times n} \tag{10}$$

Step 4: Computation of Weight-Cost matrix

The Weight-Cost matrix is computed using a normalized matrix and weights of the attribute. The Eq. (11) shows the Weight-Cost computation of y^{th} attribute of the node at x^{th} row.

$$WC^i[x][y] = R^i[x][y] * W^i[y] \tag{11}$$

The Weight-Cost matrix is represented as shown in Eq. (12).

$$(WC^i[x][y])_{m\times n} \tag{12}$$

Step 5: Computation of positive and negative ideal solutions for each attribute:

$$\begin{aligned} &A^{i+} = \{A[1]^+, A[2]^+, A[3]^+, \ldots, A[n]^+\} \\ &\text{where} A[y]^+ = \{MAX(WC^i[x][y]),\ \forall y \in \text{ benefit attributes}\}\ \&\&\ MIN(WC^i[x][y]), \\ &\forall y \in \text{ cost attributes}\} \end{aligned} \tag{13}$$

$$\begin{aligned} &A^{i-} = \{A[1]^-, A[2]^-, A[3]^-, \ldots, A[n]^-\} \text{ where } A[y]^- \\ &= \{MIN(WC^i[x][y]),\ \forall\, y \in \text{benefit attributes}\ \&\&\ MAX(WC^i[x][y]),\ \forall\, y \in \text{ costattributes}\} \end{aligned} \tag{14}$$

The positive-ideal solution maximizes the benefit and minimizes the cost criteria. The positive ideal solution $A[y]^+$ of attribute y is calculated as per Eq. (13). The negative-ideal solution maximizes the cost criteria and minimizes the benefit criteria. The negative ideal solution $A[y]^-$ of attribute y is calculated as per Eq. (14). In short, the positive-ideal solution is made up of all the best possible criteria values. In contrast, the negative-ideal solution comprises of all the worst possible criteria values.

Step 6: Finding Euclidean distance from positive and negative ideal solutions for each node in PRN

Euclidean distance of node at x^{th} row, from positive ideal solution $ED^i[x]^+$ is obtained by using Eq. (15).

$$ED^i[x]^+ = \sqrt{\sum_{y=1}^{n}(WC^i[x][y] - A[y]^+)^2} \tag{15}$$

Matrix consists of Euclidean distance from the positive ideal solutions are represented by Eq. (16)

$$(ED^i[x]^+)_{m\times 1} \tag{16}$$

Euclidean distance of node at x^{th} row, from negative ideal solution $ED^i[x]^-$ is obtained by using Eq. (17).

$$ED^i[x]^- = \sqrt{\sum_{y=1}^{n}(WC^i[x][y] - A[y]^-)^2} \tag{17}$$

Matrix consists of Euclidean distance from the negative ideal solutions are represented by Eq. (18)

$$(ED^i[x]^-)_{m\times 1} \tag{18}$$

Step 7: Score computation of each node in PRNs:

Score of node correspond to x^{th} row is computed by the Eq. (19).

$$Score^i[x] = \frac{ED^i[x]^-}{ED^i[x]^- + ED^i[x]^+} \tag{19}$$

The matrix consists of scores of all nodes represented, as shown in Eq. (20). The node that has a high score indicates higher priority and is more preferred.

$$(Score^i[x])_{m\times 1} \tag{20}$$

In the state-of-the-art HydroCast, which is used to compare with CMAR, the suitability of neighboring nodes is measured by computing the Normalized Advancement (NADV) of the nodes present at the lower depth than the source/forwarder i. The NADV is calculated by Eq. (21) [14].

$$NADV_{i,j} = ADV_{i,j} \times P_{i,j} \tag{21}$$

where $NADV_{i,j}$ is NADV of neighbor j deployed at lower depth with reference to source/forwarder i. $ADV_{i,j}$ is the advancement of the node j with reference to source/forwarder i and it is the depth difference between the source/forwarder i with node j. $P_{i,j}$ is packet delivery probability between source/forwarder i and j [14].

3.2.2 Cluster Formation

Algorithm 1 Cluster formation in CMAR at the node i

1: **Initialization:**
2: Number of clustered nodes of i $(NC_i) \rightarrow 0$
3: $PRN_i = \{j_1, j_2, j_3, ...j_p\}$, where priority of $j_1 > j_2 > j_3 > ... > j_p$
4: Threshold number of nodes to be clustered (TN)
5: k^{th} cluster of node i $(Cluster_{i,k})$; $k \rightarrow 1$
6: First node in the cluster CH
7: Distance between CH and $node$ $(D_{CH,node})$
8: l^{th} index of the node in PRN_i is not included in any of the cluster and it is represented by $V[l] = 0$
9: **End of Initialization**
10: **for** $(l = 1\ to\ \mid PRN_i \mid)$ **do**
11: $\quad V[l] = 0$
12: **end for**
13: **while** $(NC_i < TN\ \&\&\ \mid NC_i \mid \neq \mid PRN_i \mid)$ **do**
14: $\quad$ **for** $(l = 1\ to\ \mid PRN_i \mid)$ **do**
15: $\quad\quad$ **if** $(V[l] == 0)$ **then**
16: $\quad\quad\quad Cluster_{i,k} \leftarrow PRN_i[l]$;
17: $\quad\quad\quad CH \leftarrow PRN_i[l]$;
18: $\quad\quad\quad V[l] = 1$;
19: $\quad\quad\quad break$;
20: $\quad\quad$ **end if**
21: $\quad$ **end for**
22: $\quad$ **for** $(l = (l+1)\ to \mid PRN_i \mid)$ **do**
23: $\quad\quad node = PRN_i[l]$
24: $\quad\quad$ **if** $(D_{CH,node} \leq \frac{Range}{2} \&\&\ V[l] == 0)$ **then**
25: $\quad\quad\quad Cluster_{i,k} \leftarrow Cluster_{i,k} \cup \{node\}$
26: $\quad\quad\quad V[l] = 1$
27: $\quad\quad$ **end if**
28: $\quad$ **end for**
29: $\quad k = k + 1$
30: $\quad NC_i = NC_i + \mid Cluster_{i,k} \mid$
31: **end while**

Once the scores of PRN_i are computed, the next step in CMAR is determining the candidate forwarding set, by forming cluster(s) of suitable nodes. The clustering approach used in the CMAR requires only the Threshold Number (*TN*) of nodes from PRN_i to be clustered. Once *TN* number of nodes are clustered, the clustering process is terminated.

The process of clustering in the CMAR is described in Algorithm 1. The clustering process is initiated by the source/forwarder *i* (line no. 1 - 9). The clustering process begins with a node that has the highest priority and is not a part of any cluster; that node is referred to as *CH*. Additionally, node *CH* is marked as included in the cluster (line no. 14 - 21).

Further, i obtains the neighbors of node CH, which are at Euclidean distance $\leq (\frac{Range}{2})$, where $Range$ is the coverage area. Finally, node CH, along with common nodes (not part of any cluster) in the PRN_i and neighbors of node CH at the Euclidean distance $\leq (\frac{Range}{2})$ are included in the cluster (line no. 22 - 28). The cluster formation process is repeated until the TN number of nodes is included in the cluster (line no. 13). In this way, node i generates its clusters ($Cluster_{i,1}$, $Cluster_{i,2}$,..., $Cluster_{i,k}$).

The cluster formation process with TN = 4, is explained with the example shown in Fig. 3. The node i is the source/forwarder node. The node i computes its PRN_i by using Eq. (4) and subsequently determines their priority (or score), assuming that $PRN_i = \{j_1, j_2, j_3, j_4, j_5, j_6\}$, are arranged in decreasing order of their priority using TOPSIS. Further, using the Algorithm 1, node i selects the highest priority node j_1 as the CH and includes it in its first cluster ($cluster_{i,1}$). Additionally, node i finds nodes that are common in its PRN_i and nodes that are at a Euclidean distance of $(\frac{Range}{2})$ of CH (using two-hop information). Accordingly, j_3 and j_5 are included in $Cluster_{i,1}$. Moreover, no more nodes are exist in the $(\frac{Range}{2})$ of j_1 and $Cluster_{i,1} = \{j_1, j_3, j_5\}$. However, only 3 nodes are included in the cluster, and $TN = 4$; thus, further clustering continues. Nodes j_2, j_4, j_6 are not part of any cluster, however, among these nodes, j_2 has the highest priority cluster. Further, node i selects j_2 as CH in second cluster ($Cluster_{i,2}$) and subsequently includes j_4 in it. Further, no more nodes exists in the $(\frac{Range}{2})$ of j_2 and $Cluster_{i,2} = \{j_2, j_4\}$. Thus, the total number of nodes included in the cluster is 5 (in both $Cluster_{i,1}$ and $Cluster_{i,2}$), satisfying $TN = 4$. Node i forwards the data to nodes of the first cluster ($Cluster_{i,1}$). In the case of none of the nodes in the first cluster receiving the transmitted data from node i, data is forwarded to the second cluster ($Cluster_{i,2}$). Likewise, in the case of none of the nodes in the $Cluster_{i,c}$ receive the data transmitted from source/forwarder i, it will forward the data to $Cluster_{i,c+1}$.

The state-of-the-art HydroCast routing protocol, which is used to compare with CMAR, uses Normalized Advancement (NADV) of neighbors deployed at the lower depth. Also, NADV is used to create a cluster(s). The major difference between the CMAR and

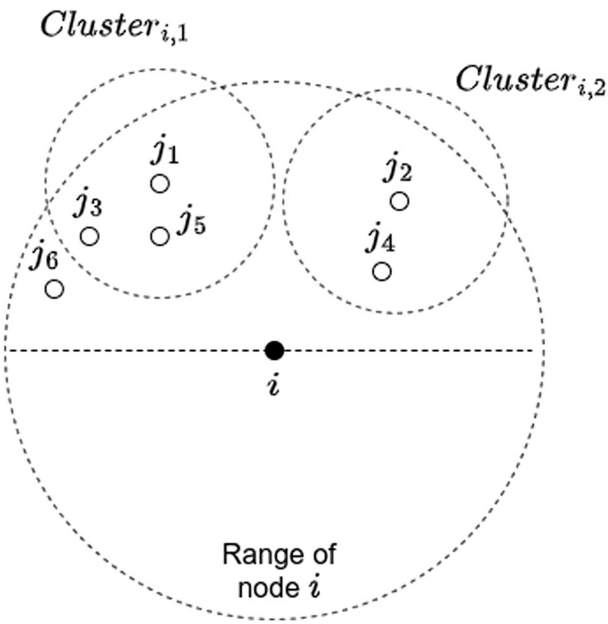

Fig. 3 Example of the cluster formation of node i in the CMAR with $TN = 4$

HydroCast is the inclusion of neighbor nodes into the cluster(s). In CMAR, only the PRNs of the source/forwarder node are considered; thereby, it avoids the inclusion of void nodes into the cluster. Further, in CMAR, only *TN* number of nodes are clustered. However, in HydroCast status of the neighbors is not considered. Void nodes can also be part of the clusters, and subsequently, they may become part of the candidate forwarding set. Further, HydroCast requires all neighbor nodes to be clustered and the Expected Packet Advancement (EPA) of individual clusters is computed by using Eq. (22).

$$EPA(S_l) = \sum_{j=1}^{l} ADV_{i,j} P_{i,j} \prod_{k=0}^{j-1} (1 - P_{i,k}) \tag{22}$$

where S_l is the set nodes of cluster having priorities $n_1 > n_2 >... > n_l$, $ADV_{i,j}$ and $P_{i,j}$ are the advancement and packet delivery probability of node j, with reference to source/forwarder i. Finally, a cluster that has the highest EPA is selected as its candidate forwarding set. In conclusion, in most cases, HydroCast results in the formation of more clusters than CMAR. Thus, HydroCast increases the complexity of cluster formation. However, only *TN* number of nodes in the CMAR are clustered. The *TN* number of nodes is adjusted based on the underwater condition/application.

3.3 Phase-II: Data Forwarding

This section elaborates on the mechanism used in forwarding the received data in the CMAR. Once data is delivered to the cluster, further forwarding of data by nodes of the respective cluster is coordinated using hold time computation. Every node j belonging to the respective cluster, which receives the data, computes its hold time T^j_{HOLD}. Hold time defines the duration of time the packet is held by the node j before it is further forwarded to its cluster. If a node j overhears the transmission of the same data packet during the hold time, it discards the packet from its buffer, to prevent duplicate transmission. In case the transmission of a packet is not overheard before the expiry of the hold time T^j_{HOLD}, no other nodes forward the same data packet. Thus node j forwards the packet to its cluster(s). The hold time T^j_{HOLD} computation for node j is given in the following section.

3.3.1 Hold Time Computation

The hold time computation by every node j in the respective cluster, which receives the data, is computed using Eq. 23).

$$T^j_{HOLD} = T^j_{PAUSE} + T^j_{OVERHEAR} + T^j_{TRANS} + T^j_{PROC} \tag{23}$$

where T^j_{HOLD} is the hold time of the node j in the corresponding cluster, T^j_{PAUSE} is the pause period of the node j, which is used to achieve synchronization among all nodes in the cluster. Further, lower priority nodes, within the cluster, delay their packet forwarding by overhearing time $T^j_{OVERHEAR}$, transmission time T^j_{TRANS}, and processing time T^j_{PROC}. The pause time of node j (T^j_{PAUSE}) is defined as in Eq. (24).

$$T^j_{PAUSE} = \frac{Range - D_{i,j}}{V} \tag{24}$$

where V is the propagation speed of the acoustic signal underwater. The total overhearing time of the node j ($T^j_{OVERHEAR}$) is calculated as per the Eq. (25).

$$T^j_{OVERHEAR} = (P_j - 1) \times \frac{Range}{V} \tag{25}$$

where P_j is the priority of the node j in the corresponding cluster. Further, total transmission time of the node j (T^j_{TRANS}) is calculated as per the Eq. (26).

$$T^j_{TRANS} = (P_j - 1) \times T_{TRANS} \tag{26}$$

where T_{TRANS} is the transmission time of a packet at the node. The total processing time of node j (T^j_{PROC}) is calculated as per the Eq. (27).

$$T^j_{PROC} = P_j \times T_{PROC} \tag{27}$$

where T_{PROC} is the packet processing delay at the node. As per the Eq. (23) through (27), a higher priority node has a lower hold time, and a lower priority node has a higher hold time.

3.3.2 Forwarding of Data

Once node j receives the packet, it has to forward it further, when the hold time T^j_{HOLD} expires. However, there are some situations in which the *TN* nodes of source/forwarder j may not be present in a single cluster and are distributed among more than one cluster. In such cases, if nodes in the first cluster fail to receive a data packet, the forwarder j forwards it to the next cluster. The overall process of the data packet forwarding is shown in Fig. 4. The packet forwarding scenarios are explained as follows:

- *Scenario 1: All TN nodes are in single cluster of source/forwarder j*
  If *TN* nodes are present in the single cluster, the forwarder j forwards the data packet to its cluster and upon receiving, a node in the cluster further forwards the data packet. If no nodes in the cluster receive the data, the transmission fails.
- *Scenario 2: TN nodes are distributed among more than one cluster of source/forwarder j*
  There are situations where *TN* nodes of j are distributed among more than one cluster. In such cases, data packet forwarding takes place in the following way. Initially, the data packet is sent to the nodes of the first cluster ($Cluster_{j,1}$). The forwarding node j confirms the data reception by the cluster nodes using overhearing for further forwarding. Therefore, the forwarder j waits for the $Timeout_{j,k}$ period to overhear further forwarding. The $Timeout_{j,k}$ is given in Eq. (28).

$$Timeout_{j,k} = 2 \times \frac{R}{V} + T^l_{HOLD} \tag{28}$$

  where T^l_{HOLD} is the hold time of the least priority node l of k^{th} cluster of j. During $Timeout_{j,k}$ duration, forwarding node j fails to overhear the further forwarding by any of the nodes of $Cluster_{j,1}$, and it is sent to its next cluster ($Cluster_{j,2}$). The same process

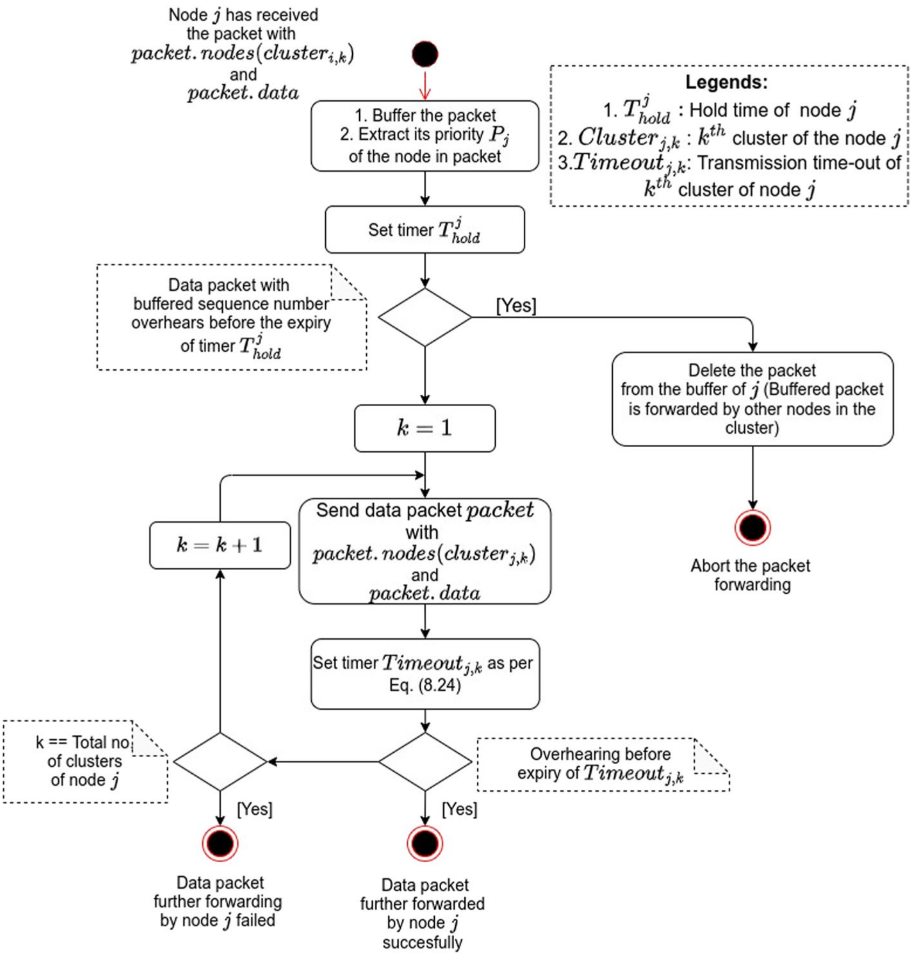

Fig. 4 Packet reception and forwarding by node j in CMAR

continues until the forwarding node j overhears the transmission or all clusters of j fail to forward the data further.

4 Implementation of CMAR

This section elaborates implementation details of the CMAR. The CMAR is implemented using MATLAB. Even though CMAR can consider any number of attributes, to consider similar attributes used in the state-of-the HydroCast, we are using progress or advancement of the neighboring nodes and packet delivery probability with neighboring nodes. The Sect. 4.1 describes the calculation of the progress or advancement attribute of the neighbor. Section 4.2 elaborates calculation of the packet delivery probability in the CMAR.

4.1 Progress of the Neighboring Node with Respect to the Source

$$PROG(n_j, n_i) = \{D(n_i, d) - D(n_j, d), \forall j \in N_i\} \tag{29}$$

When a node has a packet to forward, it will consider the progress of the neighboring nodes as one of the attributes. The progress of the neighbor j with reference to the source or forwarding node i is calculated by the Eq. (29). Neighbor with higher progress, a more preferred node to be a part of the candidate forwarding set. The maximum progress of the neighboring node j with respect to the source node i is equal to the communication range (R). Thus the progress of the node j varies from 0 to R. As shown in Fig. 5, communication range (R) is equal to 1000 ms. For the given neighboring node, progress is considered as 200, 400, 600, and 800. It means that each node considers 4 different values of progress. Suppose there are 10 neighboring nodes. They generate 4^{10} distinct combinations of progress.

4.2 Packet Delivery Probability

This subsection describes the method to determining the packet delivery probability $p(d, m)$ between a pair of nodes, which are separated by distance d for m bit transmission in the underwater. Further, discuss about the how Packet Delivery Probability (PDP) used in the CMAR. PDP is given by the the equation as follows:

$$p(d, m) = (1 - p_e(d))^m \tag{30}$$

In the Eq. (30), $p_e(d)$ is the probability of the bit error over a distance d and it is calculated by the Eq. (31).

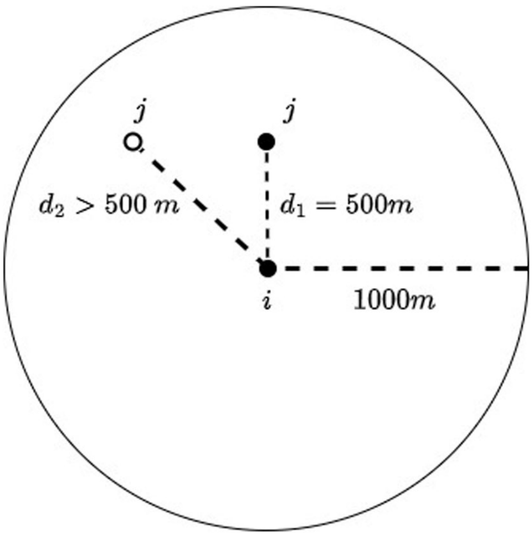

Fig. 5 Variation of the Packet delivery probability with progress

$$p_e(d) = \frac{1}{2}\left(1 - \sqrt{\frac{SNR(d)}{1 + SNR(d)}}\right) \tag{31}$$

where $SNR(d)$ is the signal-to-noise ration over the distance d. The $SNR(d)$ is computed by the Eq. (32).

$$SNR(d) = \frac{E_{bit}}{N_0 d^k a(f)^d} \tag{32}$$

In the Eq. (32), E_{bit} is the average transmission energy per bit, N_0 is noise power density in a nonfading additive white Gaussian noise (AWGN) channel, k is the spreading factor and $a(f)$ is the absorption coefficient $a(f)$ in dB/km for the frequency f in kHz. $a(f)$ is computed by Thorp's equation and it is calculated by Eq. (33)

$$10loga(f) = 0.11\frac{f^2}{1+f^2} + 44\frac{f^2}{4100+f^2} + 2.7510^{-4}f^2 + 0.003 \tag{33}$$

The packet delivery probability is computed by using the Eqs. (30) through (33). Consider the Fig. 5, when neighbor node n_j at the progress of 500 ms, with respect to the sender n_i, it indicates that the minimum distance between them is 500 ms. This is because progress is the minimum distance between the sender and the neighbor. The actual distance is anything between 500 and1000 ms. Thus for every combination of progress, two combinations of PDP's are generated as follows:

- Combination 1: By computing PDP of the corresponding neighbor by considering distance which is the same as progress.
- Combination 2: By halving the PDP of every node obtained in combination 1.

In this way, for 4^{10} advancement combinations and for each advancement combination, 2 PDP combinations are generated. Thus totally the experiment is carried out for $2 \times 4^{10} = 2,097,152$ combinations. Thus exhaustive simulations are carried out. Further, to analyze the performance of CMAR thoroughly, we varied TN from 3 to 8 nodes. The performance of CMAR is compared with HydroCast in terms of the number of forwarding nodes selected, % input combinations number of clusters formed, transmission reliability in % input combinations, % input combinations nodes at priority 1 & 2. Further, for the computation of the transmission reliability, the packet delivery probability (PDP) from the neighbor node to the sink is taken as follows:

1. PDP from void-node to the sink is taken as 0.6 times that of the source to the corresponding void-node. If the packet is stuck at the void-node, recovering from it and delivering the packet to the sink results in a higher number of hops. Thus it results in lower PDP.
2. PDP from the normal node to the sink is taken as 0.75 times that of the source to the corresponding node. If a packet passes through normal nodes, it results in a lower number of hops than void-node recovery. Thus we considered higher PDP than that of void-node.

5 Results and Analysis

The section presents the simulation results of CMAR, and it is compared with the state-of-the-art HydroCast. Section 5.1 presents the simulation parameters used for the evaluation of CMAR and HydroCast. Section 5.2 presents the detailed analysis of the performance of the CMAR in comparison with the HydroCast.

5.1 Simulation Setup

The implementation of CMAR and HydroCast is carried out using MATLAB. MATLAB has emerged as one of the appropriate network simulation tools along with ns2, ns3, QualNet, UnetStack, etc., because of its simple syntax and easier development through the advanced library. According to the survey, many of the underwater communication contributions are been justified using MATLAB [9, 37–41].

The simulation parameters are mentioned in Table 1. There is a source or forwarding node and 10 neighboring nodes in the simulation. Out of 10 neighboring nodes, any 2 nodes are selected as void nodes. They have remained as void-node until the completion of the simulation. As mentioned in Sect. 4, CMAR considers 2 attributes, the advancement and Packet Delivery Probability (PDP). With the intention of performing exhaustive simulations, 2,097,152 combinations of inputs are generated and executed.

5.2 Simulation Results

In this subsection, the CMAR is compared with HydroCast in terms of the number of forwarding nodes selected, number of clusters formed, Average EPA, Nodes priority, and transmission reliability.

5.2.1 Number of Forwarding Nodes Selected

This subsection shows the number of forwarding nodes selected in the CMAR and HydroCast for % of input combinations. The number of forwarding nodes selected indicates the potential of the source node to forward the data successfully. Table 2 shows the number of nodes in the selected cluster for CMAR and HydroCast with % of input combinations. As more nodes are present in the selected cluster, there are high chances of successful data forwarding. In the case of CMAR with TN = 3 through 8, there are high % of times 4 or more

Table 1 Simulation parameters

S. No.	Parameters	Value
1	No. of neighbors	10
2	No. of void nodes	2
3	Communication range	1000
4	Variation of depth parameter	200–800 in the step of 200
5	Packet size	1600 bits

Table 2 Number of nodes in the selected cluster(s) in the CMAR and HydroCast

No. of nodes in the selected cluster(s)*	CMAR with Threshold Number (TN) of nodes						HydroCast
	TN = 3	TN = 4	TN = 5	TN = 6	TN = 7	TN = 8	
	% of input combinations						
1	0	0	0	0	0	0	0.04
2	0	0	0	0	0	0	9.21
3	43.85	0	0	0	0	0	36.77
4	34.55	48.48	0	0	0	0	33.31
5	16.31	34.78	55.28	0	0	0	15.06
6	4.54	13.77	34.20	65.33	0	0	4.51
7	0.68	2.73	9.54	30.25	79.22	0	0.93
8	0.04	0.21	0.97	4.41	20.77	100	0.12
9	0	0	0	0	0	0	0.01
10	0	0	0	0	0	0	0.00
Total	99.97	99.97	99.99	99.99	99.99	100	99.96

*In the HydroCast, total nodes in the finally selected cluster, whereas other nodes in the other clusters are not included. In the case of CMAR, other nodes not in the cluster

Table 3 Various scenarios of cluster formation in CMAR with TN = 3

Scenario	No. of nodes in the selected cluster	Cluster 1	Cluster 2	Cluster 3
1	3	3	–	–
2	3	1	1	1
3	3	1	2	–
4	3	2	1	–
5	4	4	–	–
6	4	1	1	2
7	5	1	1	3
8	6	1	1	4
9	7	1	1	5
10	8	1	1	6

than 4 nodes are selected compared to HydroCast. It shows that CMAR provides increased chances of successful packet forwarding from the sender to its neighboring nodes.

As shown in Table 2, for example, in the CMAR, even though TN = 3, the number of nodes in the selected cluster(s) is 3 or even more. Consider TN = 3, a maximum of 3 clusters can be created considering 1 node each in the 3 clusters, and a minimum 1 cluster is created so that all 3 may present in a single cluster itself. Various scenarios for the selection of nodes in the case of TN = 3 are shown in Table 3.

As mentioned in Table 3, suppose the number of selected nodes is 3; there are 4 possible scenarios. In scenario 1, all 3 nodes are in the first cluster, the number of nodes chosen fulfills the threshold number (TN) requirements. Thus, the clustering process stops. In scenario 2, the first cluster has only one node; therefore, it will continue clustering until a minimum of 3 nodes are clustered. Even the second cluster has 1 node; thus, it looks for the third cluster and has one node. In this way, 3 nodes are distributed in 3 different clusters, as shown in scenario 2 of Table 3. Similarly, in scenarios 3 and 4, the first cluster has 1 node, and the second cluster has 2 nodes and vice-versa. The important aspect in CMAR is even though TN = 3, even more than 3 nodes are also selected in the cluster. Consider scenario 5, in which 4 nodes are selected, and this is because the first cluster itself has 4 clusters as shown in Table 3. In scenario 6, the first 2 clusters have 2 nodes. Minimum 3 nodes must be selected. Thus it will make the third cluster, and it has 2 nodes. Thus 4 nodes are chosen from 3 clusters even though TN = 3. Likewise, even though TN = 3, there are chances of even 5, 6, 7, and 8 nodes being selected as shown in scenario 7 through 10 in Table 3. However 9 or 10 nodes will not be selected because 2 nodes are void. The CMAR completely avoids void nodes, resulting in CMAR selecting the maximum of 8 nodes.

5.2.2 Number of Clusters Formed

This subsection elaborates the number of clusters formed in the CMAR for TN ranges from 3 to 8 nodes and HydroCast. As more clusters are formed, the number of nodes in the individual cluster decreases. Thus the construction of a minimum cluster is desired.

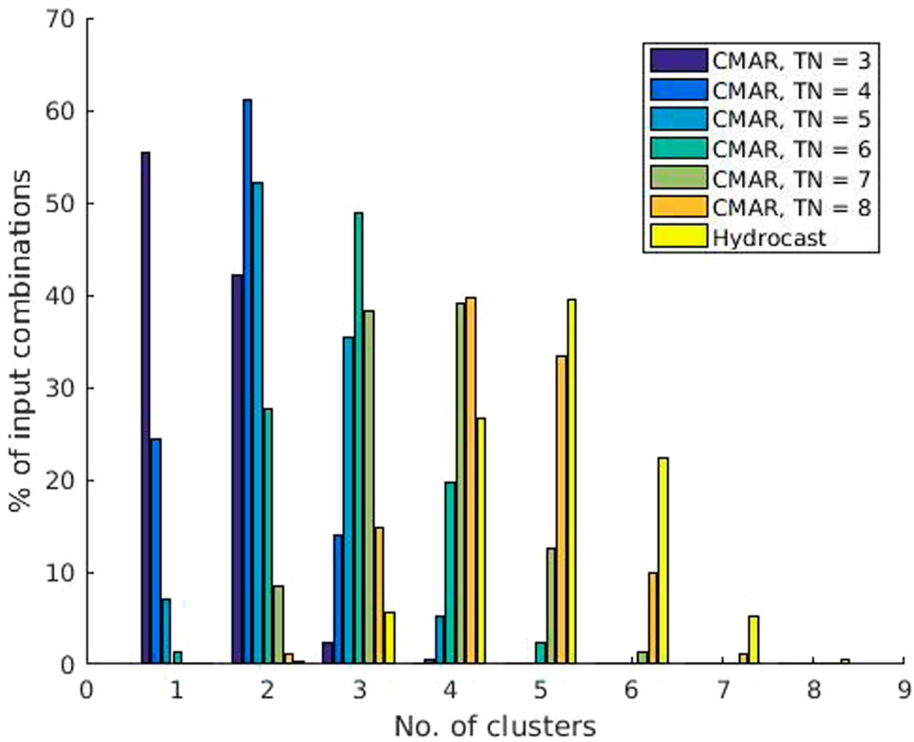

Fig. 6 Comparison of CMAR and HydroCast w.r.t. chances of forwarding packets

The number of clusters formed in CMAR with TN = 3 to 8 and HydroCast is shown in Fig. 6. In both CMAR and HydroCast, in the best case, 1 cluster is formed. In the worst case, CMAR with TN = 3, 3 clusters are possible because cluster 1, 2, 3 contains 1 node each. In this way, in the worst case, CMAR with TN = 4, 5, 6, 7, 8 requires 4, 5, 6, 7, 8 clusters, respectively. In CMAR number of clusters formed and nodes selected depends on the Threshold Number (TN). However, in the HydroCast, each node must be part of one cluster.

$$N_{max} = N - C + 1 \tag{34}$$

In HydroCast, the Eq. (34) gives the maximum number of nodes part of the single cluster. The N_{max} is the maximum number of nodes in the single cluster, N is the total number of nodes present, C is the total number of clusters formed. For example, if there are 10 nodes and 5 clusters are formed in HydroCast, the maximum number of nodes present in the single cluster will be 6 as per Eq. (34). Similarly, for 6, 7, 8, 9, and 10 clusters, the maximum number of nodes present in the single cluster will be 5, 4, 3, 2, 1, respectively. It indicates that, as the number of clusters increases, the maximum number of nodes present in the single cluster decreases in the HydroCast. The reduced number of nodes in the cluster results in lower chances of successful transmission with its neighbor nodes.

In almost 95% of input combinations, forwarding nodes are accommodated within 1 to 3 clusters for CMAR with TN = 3. Similarly, in 97% of input combinations, CMAR with TN = 4 is accommodated in just 1 to 3 clusters. However, in just 6% of input combinations, HydroCast is accommodated with 3 clusters, and in 65% of input combinations, HydroCast forms 5 to 8 clusters. As explained in Sect. 3.2.2, in CMAR clustering process is initiated from a neighboring node having the highest score. Once a cluster is formed, nodes present in it will become part of the forwarding set, and the clustering process will be stopped once the Threshold Nodes (TN) of nodes are reached. Thus in CMAR, all neighboring nodes need not be clustered. It results in a lower number of clusters. However, in the case of HydroCast, clustering is done until every node is part of the cluster. Further, by computing the EPA of each cluster, the best cluster is shortlisted, and nodes present in it will be forwarding nodes.

5.2.3 Average Expected Packet Advancement (EPA)

EPA is the normalized sum of advancements made by the selected nodes in the cluster. EPA describes how much closer a packet can be forwarded towards the destination during transmission. We compute the EPA to maximize the chance of successful delivery of a packet. The EPA is computed by the Eq. 22). EPA of CMAR with TN = 3 to 8 and HydroCast is given in Fig. 7. As discussed in the Sect. 5.2.1, CMAR results in more forwarding nodes than HydroCast. Thus, CMAR also results in a higher average EPA compared to HydroCast.

5.2.4 Nodes at Priority 1 and 2

This subsection discusses %ge of input combinations individual nodes are at priority 1 and 2. Further, elaborate its impact on the performance. Figure 8a, b, shows %ge of input combinations nodes are at priority 1 and 2. As shown in Fig. 8a, b, node Id 9 & 10 are not selected at all as priority 1 or priority 2 nodes in CMAR. However, in the case of HydroCast, nodes Id 9 and 10 are selected as priority 1 & 2 nodes even though they are void nodes.

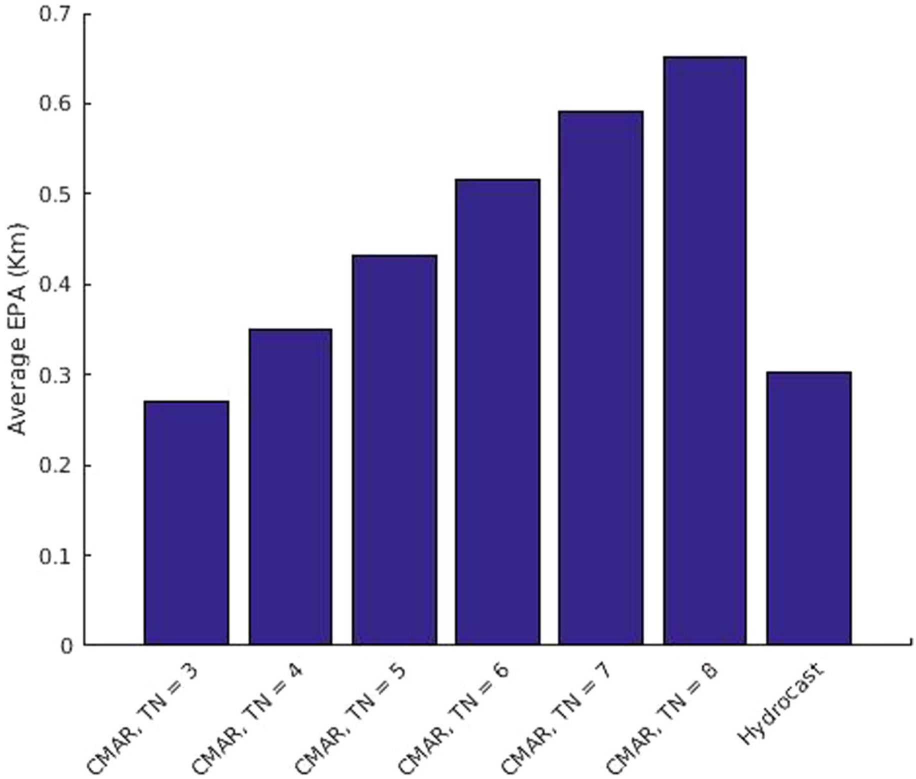

Fig. 7 Average EPA of CMAR & HydroCast

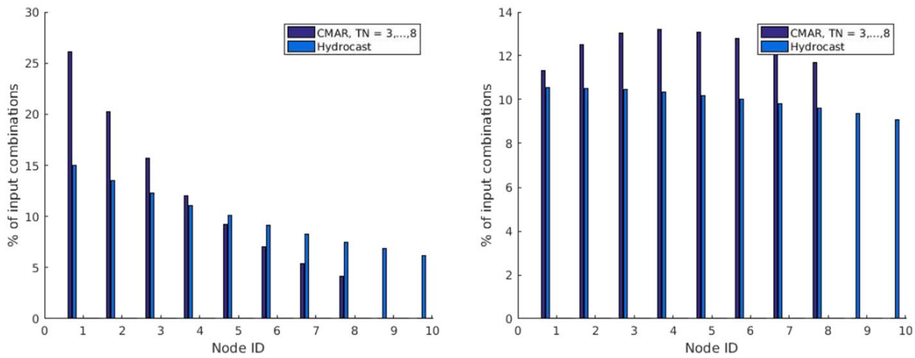

Fig. 8 % input combinations **a** Nodes at priority 1 (left) **b** Nodes at priority 2 (right)

We can conclude that CMAR completely prevents the void nodes from becoming cluster members. Thereby void nodes are not selected as forwarding nodes. However, HydroCast will not avoid void nodes. The involvement of void nodes during routing has severely impacted the network's performance.

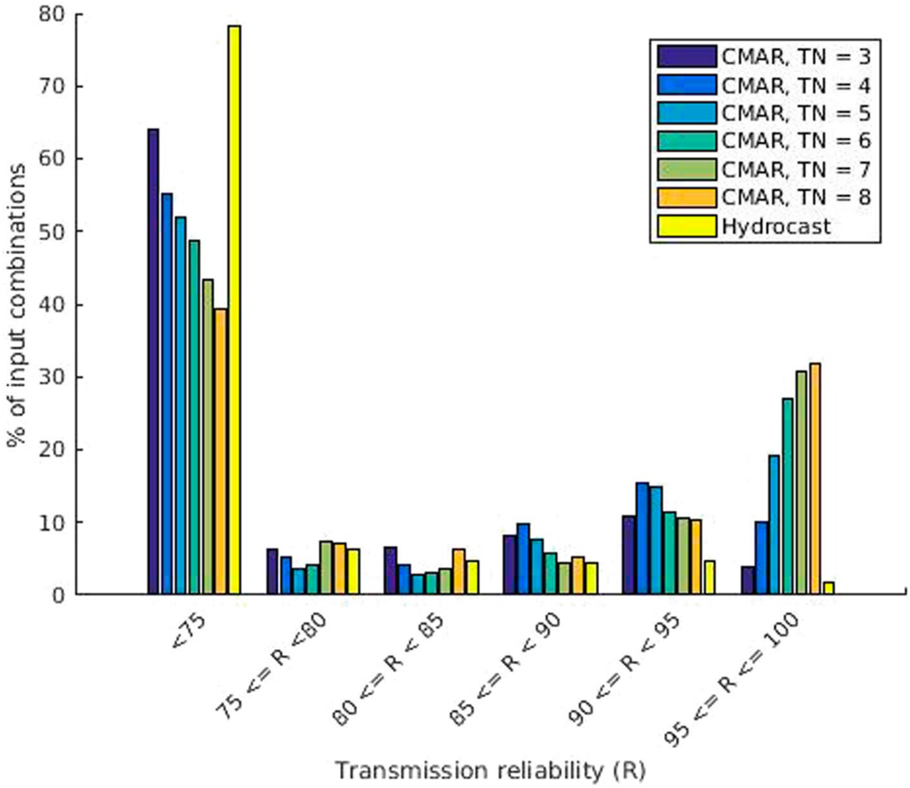

Fig. 9 Transmission reliability vs % of input combinations

5.2.5 Transmission Reliability

Figure 9 shows the transmission reliability of the CMAR and HydroCast. The transmission reliability of opportunistic routing is calculated as per the Eq. (35), [42]

$$T_{OR} = (1 - ((1 - P_{F_1}) \times (1 - P_{F_2}) \times (1 - P_{F_3}) \ldots \times (1 - P_{F_N})))^n \tag{35}$$

where P_{F_1}, P_{F_2}, P_{F_3} and P_{F_N} represents the packet delivery probability between the source or forwarding node to nodes in its candidate set and n represents the number of hops packet has to travel.

6 Conclusions and Future Work

The paper proposed a CMAR protocol for underwater networks, mainly considering multiple attributes of neighboring nodes and forming the clusters. Exhaustive simulations are carried out using MATLAB to evaluate the performance of the CMAR by varying the threshold number of forwarding nodes. Further, CMAR is compared with the state-of-the-art HydroCast routing protocol in terms of the number of clusters formed, the number of forwarding nodes selected, the priority of the nodes in the cluster, and transmission reliability. As a future work, the performance of the CMAR can be thoroughly evaluated and analyzed with testbed implementation.

Funding The authors thank the Science and Engineering Research Board (SERB), Govt. of India for providing financial support (ref. no. EEQ/2018/001036).

Author Contributions Both authors contributed equally to conception, design, coding, manuscript writing, and review.

Code Availability Code is available only in specific cases.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Consent for Publication The authors have no objection to publishing the article.

References

1. Heidemann, J., Stojanovic, M., & Zorzi, M. (2012). Underwater sensor networks: Applications, advances and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 370*(1958), 158–175.
2. Pompili, D., Melodia, T., & Akyildiz, I.F.(2006). Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks. *In: Proceedings of the 12th annual international conference on mobile computing and networking,* (pp. 298–309)
3. Coutinho, R. W., Boukerche, A., Vieira, L. F., & Loureiro, A. A. (2015). A novel void node recovery paradigm for long-term underwater sensor networks. *Ad Hoc Networks, 34*, 144–156.
4. Akyildiz, I. F., Pompili, D., & Melodia, T. (2004). Challenges for efficient communication in underwater acoustic sensor networks. *ACM Sigbed Review, 1*(2), 3–8.
5. Akyildiz, I. F., Pompili, D., & Melodia, T. (2005). Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks, 3*(3), 257–279.
6. Coutinho, R.W., Vieira, L.F., & Loureiro, A.A. (2013) Movement assisted-topology control and geographic routing protocol for underwater sensor networks. In *Proceedings of the 16th ACM international conference on modeling, analysis & simulation of wireless and mobile systems*, (pp. 189–196)
7. Kheirabadi, M. T., & Mohamad, M. M. (2013). Greedy routing in underwater acoustic sensor networks: A survey. *International Journal of Distributed Sensor Networks, 9*(7), 701834.
8. Ghoreyshi, S. M., Shahrabi, A., & Boutaleb, T. (2017). Void-handling techniques for routing protocols in underwater sensor networks: Survey and challenges. *IEEE Communications Surveys & Tutorials, 19*(2), 800–827.
9. Alasarpanahi, H., Ayatollahitafti, V., & Gandomi, A. (2020). Energy-efficient void avoidance geographic routing protocol for underwater sensor networks. *International Journal of Communication Systems, 33*(6), 4218.
10. Xie, P., Zhou, Z., Peng, Z., Cui, J.-H., & Shi, Z. (2009) Void avoidance in three-dimensional mobile underwater sensor networks. In *International conference on wireless algorithms, systems, and applications*, (pp. 305–314) . Springer
11. Boukerche, A., & Darehshoorzadeh, A. (2014). Opportunistic routing in wireless networks: Models, algorithms, and classifications. *ACM Computing Surveys (CSUR), 47*(2), 1–36.
12. Coutinho, R. W. L., Boukerche, A., Vieira, L. F. M., & Loureiro, A. A. F. (2016). Design guidelines for opportunistic routing in underwater networks. *IEEE Communications Magazine, 54*(2), 40–48. https://doi.org/10.1109/MCOM.2016.7402259
13. Coutinho, R.W., Boukerche, A., Vieira, L.F., & Loureiro, A.A. (2015) Modeling and analysis of opportunistic routing in low duty-cycle underwater sensor networks. In *Proceedings of the 18th ACM international conference on modeling, analysis and simulation of wireless and mobile systems*, (pp. 125–132)

14. Noh, Y., Lee, U., Lee, S., Wang, P., Vieira, L. F., Cui, J.-H., Gerla, M., & Kim, K. (2015). Hydrocast: Pressure routing for underwater sensor networks. *IEEE Transactions on Vehicular Technology, 65*(1), 333–347.
15. Shahrabi, A., Ghoreyshi, S. M., & Boutaleb, T. (2016). A novel cooperative opportunistic routing scheme for underwater sensor networks. *Sensors, 16*(3), 297.
16. Zhu, Y., Tian, D., & Yan, F. (2020). Effectiveness of entropy weight method in decision-making. *Mathematical Problems in Engineering, 2020*, 1–5.
17. Foubert, B., & Mitton, N. (2021). RODENT: a flexible TOPSIS based routing protocol for multi-technology devices in wireless sensor networks. *ITU Journal on Future and Evolving Technologies, 2*(1).
18. Barbeau, M., Blouin, S., Cervera, G., Garcia-Alfaro, J., & Kranakis, E. (2015). Location-free link state routing for underwater acoustic sensor networks. In *2015 IEEE 28th Canadian conference on electrical and computer engineering (CCECE)*, (pp. 1544–1549) IEEE
19. Nazareth, P., & Chandavarkar, B. (2022). Location-free void avoidance routing protocol for underwater acoustic sensor networks. *Wireless Personal Communications, 123*, 1–26.
20. Ghoreyshi, S.M., Shahrabi, A., & Boutaleb, T. (2015) An inherently void avoidance routing protocol for underwater sensor networks. In: *2015 International symposium on wireless communication systems* (ISWCS)
21. Basagni, S., Petrioli, C., Petroccia, R., & Spaccini, D. (2015). CARP: A channel-aware routing protocol for underwater acoustic wireless networks. *Ad Hoc Networks, 34*, 92–104.
22. Han, G., Liu, L., Bao, N., Jiang, J., Zhang, W., & Rodrigues, J. J. (2017). AREP: An asymmetric link-based reverse routing protocol for underwater acoustic sensor networks. *Journal of Network and Computer Applications, 92*, 51–58.
23. Erol-Kantarci, M., Mouftah, H. T., & Oktug, S. (2011). A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Communications Surveys & Tutorials, 13*(3), 487–502.
24. Xie, P., Cui, J.-H., & Lao, L. (2006). Vbf: Vector-based forwarding protocol for underwater sensor networks. In *International conference on research in networking*, (pp. 1216–1221) Springer
25. Yu, H., Yao, N., & Liu, J. (2015). An adaptive routing protocol in underwater sparse acoustic sensor networks. *Ad Hoc Networks, 34*, 121–143.
26. Coutinho, R. W. L., Boukerche, A., Vieira, L. F. M., & Loureiro, A. A. F. (2016). Geographic and opportunistic routing for underwater sensor networks. *IEEE Transactions on Computers, 65*(2), 548–561. https://doi.org/10.1109/TC.2015.2423677
27. Noh, Y., Lee, U., Wang, P., Choi, B. S. C., & Gerla, M. (2012). VAPR: Void-aware pressure routing for underwater sensor networks. *IEEE Transactions on Mobile Computing, 12*(5), 895–908.
28. Rahman, M. A., Lee, Y., & Koo, I. (2017). EECOR: An energy-efficient cooperative opportunistic routing protocol for underwater acoustic sensor networks. *IEEE Access, 5*, 14119–14132. https://doi.org/10.1109/ACCESS.2017.2730233
29. Javaid, N., Majid, A., Sher, A., Khan, W. Z., & Aalsalem, M. Y. (2018). Avoiding void holes and collisions with reliable and interference-aware routing in underwater WSNS. *Sensors, 18*(9), 3038.
30. Guan, Q., Ji, F., Liu, Y., Yu, H., & Chen, W. (2019). Distance-vector-based opportunistic routing for underwater acoustic sensor networks. *IEEE Internet of Things Journal, 6*(2), 3831–3839. https://doi.org/10.1109/JIOT.2019.2891910
31. Zhang, Y., Zhang, Z., Chen, L., & Wang, X. (2021). Reinforcement learning-based opportunistic routing protocol for underwater acoustic sensor networks. *IEEE Transactions on Vehicular Technology, 70*(3), 2756–2770.
32. Behzadian, M., Otaghsara, S. K., Yazdani, M., & Ignatius, J. (2012). A state-of the-art survey of Topsis applications. *Expert Systems with applications, 39*(17), 13051–13069.
33. Chandavarkar, B. R., & Guddeti, R. M. R. (2016). Simplified and improved multiple attributes alternate ranking method for vertical handover decision in heterogeneous wireless networks. *Computer Communications, 83*, 81–97.
34. Vafaei, N., Ribeiro, R. A., & Camarinha-Matos, L. M. (2018). Data normalisation techniques in decision making: Case study with topsis method. *International Journal of Information and Decision Sciences, 10*(1), 19–38.
35. Chandavarkar, B. R., & Guddeti, R. M. R. (2015). Simplified and improved analytical hierarchy process aid for selecting candidate network in an overlay heterogeneous networks. *Wireless Personal Communications, 83*(4), 2593–2606.

36. Wang, L., & Kuo, G.-S.G. (2012). Mathematical modeling for network selection in heterogeneous wireless networks-a tutorial. *IEEE Communications Surveys & Tutorials, 15*(1), 271–292.
37. Rani, S., Ahmed, S. H., Malhotra, J., & Talwar, R. (2017). Energy efficient chain based routing protocol for underwater wireless sensor networks. *Journal of Network and Computer Applications, 92*, 42–50.
38. Thi Kim, O. T., Nguyen, V. D., & Hong, C. S. (2014). Which network simulation tool is better for simulating vehicular ad hoc network? *Proceedings of the Korean Information Science Conference*, *134*(1), 930–932.
39. Ghoreyshi, S. M., Shahrabi, A., & Boutaleb, T. (2016). A novel cooperative opportunistic routing scheme for underwater sensor networks. *Sensors, 16*(3), 297.
40. O'Rourke, M., Basha, E., Detweiler, C (2012) Multi-modal communications in underwater sensor networks using depth adjustment. In *Proceedings of the 7th international conference on underwater networks & systems*, (pp. 1–5)
41. Nazareth, P., & Chandavarkar, B. R. (2019). E-var: Enhanced void avoidance routing algorithm for underwater acoustic sensor networks. *IET Wireless Sensor Systems, 9*(6), 389–398.
42. Thyagarajan, J., & Kulanthaivelu, S. (2021). A joint hybrid corona based opportunistic routing design with quasi mobile sink for IOT based wireless sensor network. *Journal of Ambient Intelligence and Humanized Computing, 12*(1), 991–1009.

Pradeep Nazareth received his Ph.D. from National Institute of Technology Karnataka, Surathkal, India in 2023. Currently, he is working as a Associate professor in the Department of Artificial Intelligence and Machine Learning, Alva's Institute of Engineering and Technology, Moodbidri. His research interests include mobile ad-hoc networks and underwater acoustic networks.

B. R. Chandavarkar received his Ph.D. from National Institute of Technology Karnataka, Surathkal, India in 2016. Currently, he is working as an Associate Professor in the Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, India. His research interests are next-generation internet protocol, mobility management, and heterogeneous networks.