International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/195674 Volume 7, Issue 2, March – April (2020)

RESEARCH ARTICLE

EEMCCP - A Novel Architecture Protocol Design for Efficient Data Transmission in Underwater Acoustic Wireless Sensor Network

M. Usha
Department of MCA, MEASI Institute of Information Technology, Chennai, Tamil Nadu, India.
ushasathyam78@gmail.com

J. Sathiamoorthy
Department of Software Applications, Thiruthangal Nadar College, Chennai, Tamil Nadu, India.
jsathyam74@gmail.com

R. Ashween
Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Tamil Nadu, India.
ashweenr1998@gmail.com

Bhagavath Nishanth Ramakrishnan
School of Engineering and Computing, Western Sydney University, NSW, Australia.
nishanth.sachinr@gmail.com

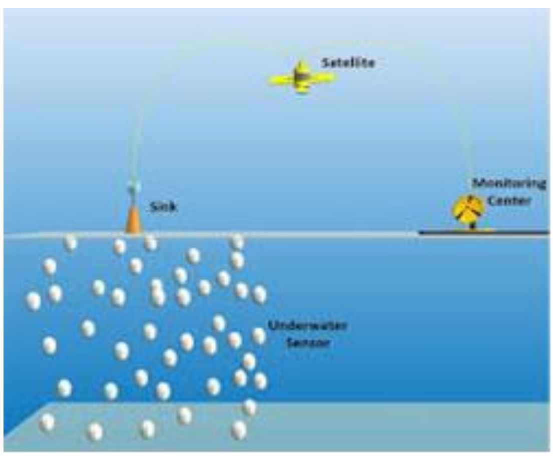
Published online: 30 April 2020

Abstract – Underwater communication is a newer addition to the ad hoc network arena as it opens several avenues of research that can be carried out under water. It has a varied range of applications which it carries out with the help of Autonomous Underwater Vehicles (AUV).Acoustic waves are used for sending out the network signals because they tend to have a lower attenuation under water. Underwater Acoustic Communication (UWAC) is used to send and receive signals underwater. Since UWAC is a relatively new field, there are still a lot of challenges to overcome. Unlike terrestrial communication, UWAC has to encounter factors like limited bandwidth, longer propagation delay, Doppler Effect, and channel availability problems. All of these factors have a direct effect on the throughput. Clustering the network can effectively help in managing the network efficiently. We have proposed an Energy Efficient Minimum Cost Cluster routing Protocol (EEMCCP), which routes packets with the help of clusters. The clustering of the network is efficiently achieved by using the Chaotic Algae Algorithm (CAA).After clustering the network, one node from each cluster acts as the Cluster Head (CH). The proposed protocol does not require any node position information and also ensures that, only a minimal number of nodes are involved in the end to end routing process. Some of the Cluster Heads are selected as gateway nodes based on their RSSI (Received Signal Strength Indictor) values of the Hello packets. Selected CHs are appointed as gateway nodes that are responsible for supplying the data to the AUV nodes that are nearer to them. When these gateway CH nodes exhaust their energy, back up nodes act as gateway nodes thus making the protocol a flexible one. In fact, the node location and position are not taken into consideration. The AUV nodes transfer the acquired information to a communication link which in turn transfers the data to a terrestrial destination. The EEMCCP focuses towards a cost effective solution for transmitting data. The performance of EEMCCP is compared with the existing protocols employed for underwater communication. From the analysis, we find that EEMCCP achieves a good throughput and PDR than the existing protocols.

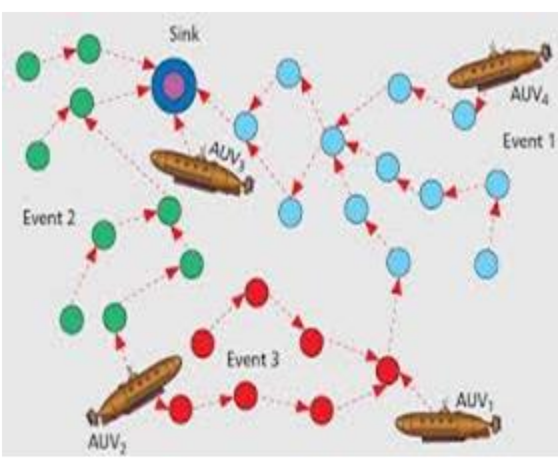
Index Terms – EEMCCP, AEERP, EEDBR, AUV, Gateway Nodes, RSSI, Routing, Cluster, Cluster Head, CAA.

1. INTRODUCTION

Underwater network communication is an advanced network where the nodes move and collect data underwater [1]. The collected data is distributed to the communication points using AUVs [2]. The sensor nodes have provisions to enable communication. In other words, the underwater nodes need to have self-set up capacities. Underwater communication has a lot of varied applications in both military and civil fields. It also helps in studying the ocean bed and the effect of environmental changes over it [3]. Underwater communication helps in collecting scientific data and supports monitoring activities based on it. The UWSNs need to exchange information regarding the configuration, location and movement of the nodes. After analyzing the data, the UWSNs transfer the data to inland points where the data is used. The nodes use radio waves to communicate their findings. Interfaces in underwater communication depend on acoustic wireless communication capabilities. Like any other network nodes, the data transmission suffers from scattering. In under water communication, the network layer helps to find the suitable path between source and destination taking into account the network features like delay, energy of the nodes. Finding routes for ad hoc WSNs are talked about in many areas. Because of the challenges in underwater environment, the difficulties need to be addressed for existing underwater networks. Existing routing protocols are categorized into proactive, reactive and geographical routing. The reason for avoiding proactive protocols in UWS networks are memory and energy [4]. But reactive protocols are unsuitable because of high latency, asymmetrical links and topology. The role Geographical based routing protocol provide localization information. Figure 1 (a & b) describes the AUVs are the communicating nodes and are responsible for the collection and distribution of data to the surface node, which in turn send it to the intended destination[5].



(a)



(b)

Figure 1 (a &b) Under Water Communication Models

The rest of the manuscript is organized as follows: Section 2 describes the latest research works that have been carried out in the underwater communication. Section 3 explains the motivation of this research work. Section 4 illustrates the challenges that exist in underwater communication. Section 5 gives the detailed demonstration of the proposed research work that have been carried out in underwater communication. Section 6 explains the various results that have been obtained in the research work. Finally section 7 concludes the research work.

2. RELATED WORK

Underwater sensor networks have a varied range of applications related to oceanography. The ocean floor is affected by changes in the environment. This has a direct effect on the marine life. Underwater communication help prevent catastrophes based on their data collection and communications capabilities .At the same time, underwater communication also suffers from issues common to networks like bandwidth, signal scattering, error rates , delay etc. Acoustic channel modeling is hot research topic and is currently being studied [6] [7] [8].The protocols pertaining to underwater communications is also studied intensively [9] [10]. UWCNs (Underwater Wireless Communication Networks (UWCNs) have sensors and AUVs to help them carry out their transmission [11]. They cater to specific applications, especially underwater monitoring (Figure 1). The nodes coordinate among themselves and share the collected data. This generally poses a security threat [12]. The oceanic floor and its environment is susceptible to adverse attacks. Inter vehicle and AUV sensor communication is quite challenging due to the mobility of the nodes. The node mobility as further hindered by the water current movement [13] [14]. Underwater communication also faces routing issues. The routing protocol must be efficient enough to route the data towards the destination with an optimal path and cost effective routing. The work in [15] studies the Travelling Salesman Problem to understand effective routing methods. The AUV can optimize the network performance by communicating with multiple nodes simultaneously. The AUV is also capable of selecting an optimal path based on contouring distance. The AUV utilizes this information for gathering data. This helps in minimizing the travelling cost and the data delivery ratio form the AUVs side. Unfortunately, this scheme incurs an elongated delay in the network because of the discrepancy of receiving information in the sensor nodes. This work focuses on implementing a cost effective path towards the end node. This scheme fails to consider an optimized data delivery ratio and delay. Multi hop routing focuses on coordination among the nodes and helps facilitate data collection [16] [17] [18].

The CEAACK MANET scheme focuses on clustering the network and vests responsibility on a cluster head. The

RESEARCH ARTICLE

scheme is an efficient one but is liable to incur overheads when the node density increases. Due to the high mobility of the nodes, the clusters may be reformed. If the cluster head moves away, the election of a cluster head is again carried out. Re-clustering and reelection processes may reduce the network performance. This would disrupt the network transmission. To address this issue the CEAACK technique introduces a Fuzzy Dynamic cluster formation scheme. Heuristic parameters of the network are utilized to aid in dynamic cluster formation [19] [20]. The node details are used to form clusters even when links are broken. The final phase of this technique ensures an optimal secure route for the transmitted data [21]. The FDCRP protocol analyzed different routing methods for MANETs. It focused on developing intelligent based routing schemes for a MANET by using an ant colony algorithm. It focuses on important parameters of an ad hoc network like dynamic cluster formation and efficient routing. Additionally, it also focuses a three tire filtering methodology to filter out the efficient nodes for the routing process. The node information is used in forming the dynamic clusters [22] [23]. This information is dynamically broadcast [24].The members of the dynamic cluster are filtered through three level process to calculate their trust index [25] [26]. This helps in filtering out the malicious nodes (even a CH) [27].A simulated analysis was made on existing protocols like AODV, OLSR and DSDV over FDCRP in order to assess its performance objectives. Routing schemes for wireless ad hoc networks are designed to efficiently utilize the scarce resources available. Battery power is a major concern and the proposed technique helps to extend the lifetime of the network within the constraints available [28-30]. The AUVs serve as relays in carrying forward the sensed information from the ocean floor [31]. The AURP deploys AUVs and gateway nodes that are predefined. The AUVs are in charge of getting the information from the CH (gateway) and delivering it to a communication point on the surface. The UWSN nodes follow a series of hops towards the gateway node. This same path is utilized throughout the communication phase. As the data size gross in size it may lead to the energy loss of the gateway node (reduce battery life) [32].This may result in the network lifetime being lessened and a minimal data delivery to the sink node.

3. MOTIVATION

Underwater communication is a challenging field with a lot of avenues yet to be explored. Routing data underwater is a major challenge since the nodes encounter the ocean water as well as aquatic flora and fauna that may be a major hindrance to overcome. The network communication range needs to be managed effectively so optimal routing can be achieved. This is achieved by clustering the network and appointing a CH for every cluster. The Cluster Head (CH) needs to be selected only after fulfilling certain qualifying criteria. These cluster heads are responsible for routing the data .Also, the cluster heads also have the tendency to move away from the cluster. Optimal cluster heads need to be identified so they can successfully transfer the data to the nearest AUV nodes. Multiple sink nodes may be deployed on the water surface which acts as destination nodes. Data received by a sink is considered as delivered data since these sink nodes have the capability of communication with nodes in the higher bandwidth range. These sink nodes deploy radio communication which ensures minimal network delay. When implementing newer techniques, their portability towards under water data exchange needs to be monitored frequently. These techniques must be able to address the network issues common to underwater communication. Again, the data forwarded underwater may be sent to a destination that exists on firmer ground (on land).Arrangements has to be made to ensure that the data travels safely towards its destination regardless of the node location. All of the protocol implementations must achieve the objectives while being cost effective as well. The data is transferred with the help of AUV nodes .These nodes need to coordinate with the CHs to transfer data to their intended destinations. The energy of the AUV nodes needs to be conserved in order to ensure effective data delivery. The proposed protocols for underwater communication needs to be scalable and energy efficient while being flexible to unforeseen issues that may arise.

4. CHALLENGES

Underwater communication faces a lot of challenges and issues which need to be addressed by the proposed protocols. EEMCCP has two types of nodes to ensure data transmission towards the destination. The ordinary nodes sense the information from the cluster heads. In turn they will transfer this information to the courier nodes which will deliver the data to the nodes on the water surface.

In our proposed work we focus on achieving the following objectives, namely,

- Cost effectiveness
- Efficient clustering of the network
- Choosing a CH
- Successful coordination of the CH with the Base Station(BS)
- Identifying the gateway cluster head nodes
- Having back up CH gateway nodes
- Collection of data and forwarding it towards the nearest AUV
- Effective energy consumption of the nodes
- Transferring data from the underwater AUV nodes to the nodes on the water surface

- Not allowing the node location to interfere with data delivery
- Maximizing network throughput and PDR
- Minimizing delay

5. PROPOSED WORK: ENERGY EFFICIENT MINIMUM COST CLUSTERING PROTOCOL (EEMCCP)

5.1. Architecture of the proposed protocol (EEMCCP)

The proposed EEMCCP scheme focuses on a competent cluster routing protocol that aims to be

- Efficient
- Dynamic
- Distributed
- Secure

Figure 2 illustrates the overall architecture of the proposed EEMCCP scheme. It gives a physical view of the underwater node setup. The diagram shows a set of Underwater Wireless Sensor Nodes (UWSN) (light blue spheres) which are scattered within the network communication range under water. The nodes are organized into clusters using the Chaotic Algae Algorithm (CAA). The Cluster Head (CH) (red spheres) is also selected by using the CAA. The selection of the CH is based on its energy level and its position within the cluster. Additionally, a second node in each cluster is selected as a Backup CH (BCH) (dark blue spheres) in the event of the original CH moving away. Once the network has been clustered , the selected CHs of every cluster coordinates with the nearest base station for routing the data .

A few of the CHs are identified as gateway CHs. The data that is sensed by the other nodes are routed to the gateway nodes. These gateway CH nodes forward the routed data to the AUV (au nodes that may be travelling inside the network communication range . The AUV nodes in turn transmit the data to the next level of communciaiton that may be situated either on the water surface(ship) or to a communicaiton point in land(a base station). The proposed EEMCCP scheme ensures safe delivery of data regardless of the location of the destination node.

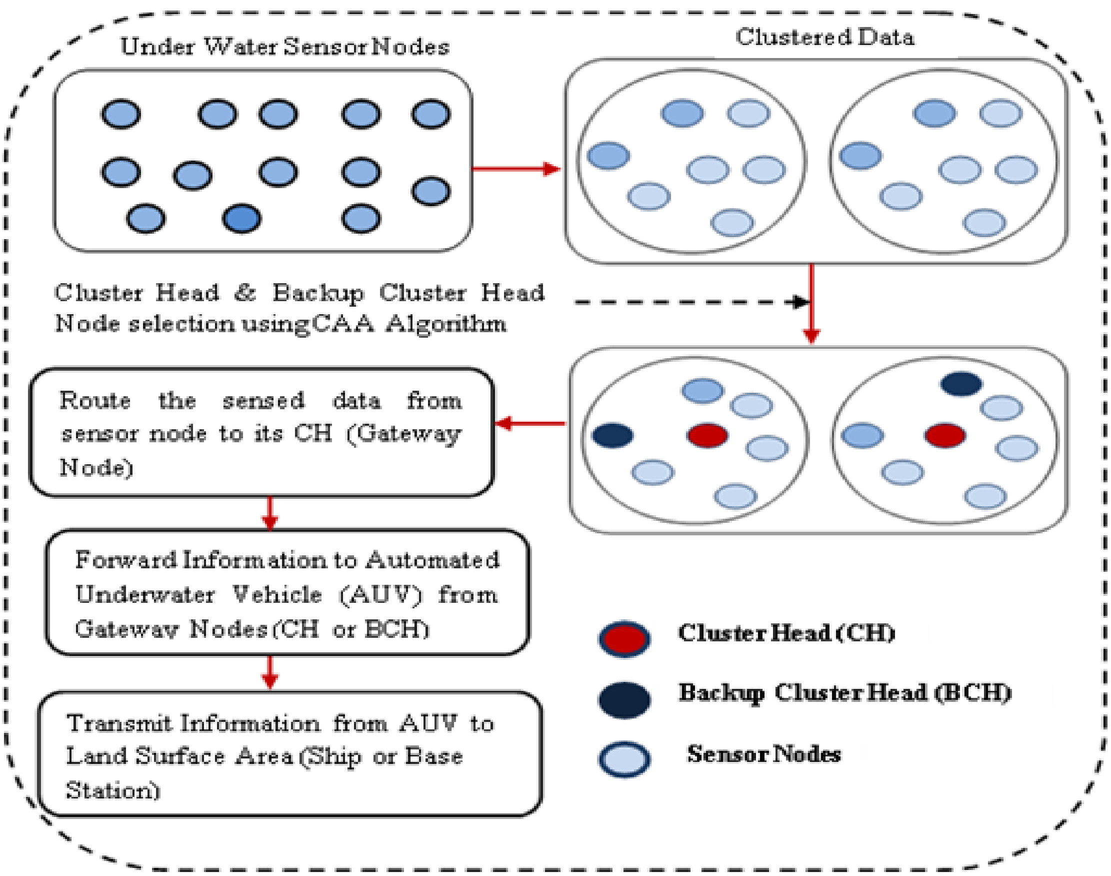


Figure 2 EEMCCP Protocol Architecture

RESEARCH ARTICLE

5.2. Phase I: Formation of Cluster and Cluster Head Selection Procedure

The EEMCCP is an energy efficient distributed cluster routing protocol which can be utilized for long- term aquatic monitoring applications using UWSNs. These UWSNs have non-periodical node mobility and do not specifically require a GPS to track them. The proposed protocol is very flexible and uses the technique of aggregation. This allows the protocol to reduce the flow of redundant information to the sink nodes responsible for sensing and collecting data under water. As stated earlier, the network UWSNs are divided into clusters and a Cluster Head (CH) is designated for each cluster. The member nodes of each cluster are expected to send their data to their CH only. Since the data within a cluster is routed only through the CH, the cluster members can send their data packets toothier respective CHs using a single hop only. In other words, all the members of a cluster can communicate via the CH only. The CH is responsible for routing the data either internally or to external CHs. All the sensor nodes in the cluster send their data (sensed data) to the CH. The CH is responsible for aggregating the data and sends it to the sink node. The CH uses the other CHs to route the data towards the sink node via multi-hops. The Cluster Heads are responsible for both intra cluster routing (routing within the members of a cluster) and inter cluster routing (routing between clusters). Since the UWSNs can deplete energy faster, the CH is selected using a randomized rotation method among each cluster. This technique helps in conserving the battery energy of the UWSNs. When the sensor nodes are deployed under water, each node is expected to send a control packet that requests for a CH .The Chaotic Algae Algorithm (CAA) initiates its work by identifying the possible CHs for the number of clusters to be formed. To carry out this task, the CAA utilizes two major parameters, namely,

- The location of the UWSN
- Energy of the UWSN(Battery power)

The Chaotic Algae algorithm is responsible for clustering the network .Whenever a sensor node is deployed and is active , it sends a control packet searching for a CH .If any of the sensor nodes accepts the hello packet it sends back an acknowledgement(ACK) to the initiating sensor node .the sensor node can then connect with the CH. If not, the nodes will enter a state of suspension (Sleep mode). A significant amount of time is spent by the sensor node in seeking out a CH. In case of not connecting with a CH after repeatedly trying for a stipulated amount of time, the sensor node enters the sleep state. This sleep state of a sensor node merely means that it will not be receiving or transmitting signals for a while, thus conserving its energy. It is interesting to note that the suspension time of a sensor node can be altered manually.

5.2.1. The Cluster Head Selection Process is Initiated as Follows

The Chaotic Algae Algorithm selects a spherical area to form a cluster with the first sink (first CH) node at the center. The maximum emission – reception distance is selected as the radii 'R 'of the spherical area **(**Figure 3**)**. As denoted in the Figure 3 CAA starts finding the CH candidates from the UWSNs that are located in the range of R/n to R. This R/n to R value denotes the range of the network is and enables the CAA to look for those sensor nodes closer to the proximity of the perimeter of the sphere.

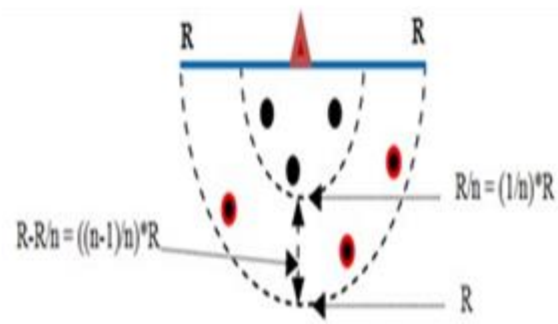


Figure 3 Clustering and Cluster Head Selection Process

From the Figure 3 we can see that the sensor nodes that are situated inside the half ring between the radii 'R/n' and 'R' are selected as the CH candidates. The remaining nodes are ordinary sensor nodes which will act as the members of the clusters. The region covered using this method depends on the value 'n' which can be maximized or minimized based on the network needs. When the first CH is chosen (say, CH1 as shown in Figure 4), the CAA now analyses the area with radius 'd' and CH1 as the center. Using this analysis, the CAA selects the sensor nodes in this range to create the second cluster (as shown in Figure 4**)**. The CAA finds the CH sensor nodes that are located in the range of R/n to R. This process is repeated until all the sensor nodes are clustered.

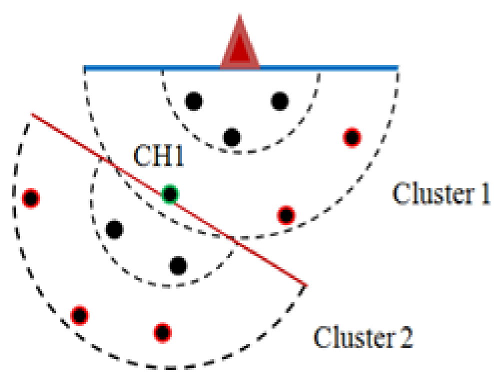


Figure 4 Back up Clustering Head Selection

RESEARCH ARTICLE

The CHs are selected mostly based on their level of residual energy. In addition to every CH selected by the CAA, an additional backup node with similar characteristics as the chosen CH is also selected. All the information that is gathered and sent to the CH is also stored in the backup node .The backup node also monitors the CH periodically in order to assess the state of the CH in terms of its energy and position within the cluster. If the CH exhibits any problem in terms of hardware, software or processing capabilities, the backup nodes assumes the CH position .This thinking ahead property of the CAA enables the network operation to go on smoothly even if the chosen CH becomes redundant at any instance. The CAA also selects the CHs for the next set of clusters for the lower tier of the network. This is achieved by the algorithm by taking the distance between the probable candidates for a CH and the CH that has been chosen already.

5.2.2. Mathematical Notation of Clustering using Chaotic Artificial Algae (CAA) Algorithm

The algae sometimes act as predators in finding their prey. This natural behavior of the algae is applied in finding the perfect nodes for each cluster. The search agent of this algorithm is responsible for finding the suitable node for a particular cluster. The search agent is also responsible for updating the positions randomly based on changes in the network. Equation (1) represents the working of the search agent of the Chaotic Artificial Algae algorithm (CAA).

Equations (2) through (4) help to decide the best agent. The factors for verifying this aspect are represented by random numbers specified in equations (3) and (4). The three vectors $\vec{U}, r, \vec{u}$ help in selecting the best node that

$$\vec{W} = \left| V . X^* \vec{(t)} - \vec{X}_p \right| \quad (1)$$

Where

t - Iteration counter

$X^*(t)$ - Best node position

$\vec{X}_p(t)$ - Vector representing node position

$$(\vec{X}_p + 1) = \vec{U} . \vec{W} \quad (2)$$

$$\vec{U} = 2\vec{u} . \vec{r} - \vec{u} \quad (3)$$

$$\vec{V} = 2\vec{r} \quad (4)$$

would join a particular cluster. The vector $\vec{u}$ helps decide whether a node can be selected for a cluster or not. When this particular vector value reaches zero, it helps in making the decision of selecting the node in consideration. $\vec{U}$ has a major role in finalizing the node to be selected. When the value of $\vec{U}$ is greater than 1, the random search agent is used and when the value is lesser than 1, the current best solution is updated.

Phase II of the Chaotic Artificial Algae algorithm identifies the best node among all the nodes to add to a cluster. The node is judged based on characteristics such as

- Proximity to the cluster
- Distance between its neighbors
- Energy level
- Latitude and longitude positions of that node within the network range
- Node stability

The cluster is thus formed with the number of nodes that satisfy the expected conditions. Algorithm 1 explains how a node is selected to join a cluster. The CAA also makes sure that none of the nodes are left out within the network range. The CAA tries to minimize the value of $\vec{u}$ from two to zero by utilizing the expression given in equation (3). Also, the value of $\vec{U}$ depends on $[-u, u]$.

The network range to be clustered is studied and the nodes are identified. As the search agent carries out the task of searching the network, the fitness function is calculated and based on the obtained values, the search position is updated. Using these values, the nodes are identified and clustered. This process is iterated until all the nodes are categorized into their respective clusters. The distance between the cluster and the nodes is expressed in equation (5). This is done to decide the membership of the node to the cluster.

$$\vec{X}_p(t+1) = \vec{W} . e^{bl} .\cos(2\pi l) + \vec{X}^*(t) \quad (5)$$

The distance $\vec{W} = \left| X^* \vec{(t)} - \vec{X}_p \right|$ represents the space interval between the node and the cluster. The probability of this distance is represented in equation (6) with random values given over the interval [-1, 1].

$$(\vec{X}_p+1)=\begin{cases} X^*(\vec{t})-\vec{W}.\vec{V} \\ D.e^{bl}.\cos(2\pi l)+\vec{X}(t)^* \end{cases} \quad (6)$$

Algorithm 1 details the formation of a cluster using the CAA technique. As explained earlier, the sensor node sets up a timer before sending out a control packet looking for a CH. This control packet is a request packet looking for a cluster head to join. If no ACK packet is received within the stipulated timer set up time, the sensor node is put to sleep (suspension mode). In case an ACK is received before the time expires, the distance between the sensor node and the Cluster Head. As specified in the CAA, the sensor node can be a cluster head or a cluster member based on the criteria satisfied by that node.

The process is repeated for all the sensor nodes till the entire network is clustered. The main characteristic of the CAA scheme is that it involves all the nodes in the communication range and organizes them efficiently into clusters. Those nodes that are unable to join a cluster are suspended to conserve energy. The formation of the clusters is also stipulated to a time limit since it would be useless if the network spends time only in clustering. As stated before, the non-members can always be suspended.

Step 1: Start

Step 2: Switch on T1 // Timer

Step 3: Set p_count // packet retry counter

Step 4: Repeat

Step 5:Send req_pkt // request packet is the control packet

Step 6:If (There no proper ACK) then // ACK is the Acknowledgement

Step 7: p_count = p_count+1

Step 8: If (p_count > Max_Trans) then

Step 9: Sensor sleep // suspension time

Step 10: End If

Step 11: Else

Step 12: Compute Distance between Head and Sensor node

Step 13: If ((Sensor_Dist <= Max_Dist) && (Sensor_Dist <= Min_Dist)) then

Step 14: Sensor_node = Cluster Head (CH)

Step 15: Else

Step 16: Sensor_node = Cluster Member (CM)

Step 17: End If

Step 18: End If

Step 19: T1 > T // T – maximum time required for cluster formation

Step 20: Stop

Algorithm 1 Formation of Clustering (CAA) Procedure

5.2.3. Data Exchange Procedure

The work of the underwater sensor nodes is to analyze the received data and gain information from the data. The two major procedures that make this possible are

- Data collection procedure
- Data transmission procedure

A communication procedure is responsible for receiving, collecting and transmitting the data. While doing this, the communication procedure must be aware of the node status where,

- The sensor node is an ordinary node responsible for gathering data from its environment
- The sensor node is the CH responsible for acquiring data both form its environment and from the other sensor node of its own cluster

Algorithm 2 illustrates the data exchange procedure .We can see that initially, the procedures responsible for data collection and transmission are invoked. If no ACK packets are forth coming it indicates that there are no clusters available .In that case, the procedure for forming the cluster is invoked thereby facilitating the network environment to carry on with its data manipulation procedure as usual. The data collection procedure and the transmit data procedures are explained in the following sections.

Step 1: Start

Step 2: Call Data Collection Procedure

Step 3: Call transmit data procedure

Step 4: If (There no proper ACK) then // ACK is the Acknowledgement

Step 5: Call Formation of Clustering Procedure

Step 6: End If

Step 7: Stop

Algorithm 2 Data Exchange Procedure

RESEARCH ARTICLE

5.2.4. Data Collection Procedure

This routine (procedure) collects data from

- The environment
- The other sensor nodes (other CHs)
- The control packets
- The ACK packets

The data thus received is checked by the following procedures namely,

- wait_For_Ack procedure - responsible for tracking the ACK packets
- wait_For_Data procedure- responsible for tracking data reception
- Check_Buffer procedure – keeps track of the buffer and over flow issues

The amount of data that is to be collected by an UWSN is largely dependent on factors like

- The size of the buffer
- Time spent by the sensor node in data collection

If the time taken to collect data exceeds the time limit , the sensor node is forced to send the data collected regardless of state that the buffer is in. this imposed condition becomes necessary since underwater data collection process is mainly used for research purposes .It becomes necessary that data is collected and sent at periodic intervals. When data collection is allowed to extend beyond its time limit, there is a high chance of the data becoming obsolete over time thus obstructing the research involved. Algorithm 3 depicts the data collection procedure. In order to avoid ambiguity, we have set a limit of 500 bytes for the buffer size. The maximum time is allotted to collect the data .If the sensor node collects data within the timer limit, the transmit procedure is invoked. If the node exceeds the timer limit, the data collected up to the timer limit is transmitted. After the collection of data, it is transmitted to the sink nodes for aggregation and analysis.

Step 1: Start

Step 2: Set Limit = 500 bytes // Buffer Limit

Step 3: Switch on T2 // T2 – Sample Timer

Step 4: Collect data from environment through nodes

Step 5: If (T2<=T) then // T: max time need to collect data

Step 6: Repeat // Iterate the process

Step 7: Call Wait_For _Data procedure

Step 8: To_Buffer > Limit

Step 9: Call Transmit Data procedure

Step 10: End If

Step 11: Stop

Algorithm 3 Data Collection Procedure

5.2.5. Transmit Data Procedure

This routine is responsible for transmitting the collected data. Depending on the nature of data collected, it is transmitted to the upper or lower tiers of the network. The data transmitted to the lower tier of a network refers to the ordinary wireless sensor nodes, while the upper tier refers to the data delivered to a CH. For instance, if the nodes are an ordinary client sensor node, the data transmitted by them could be

- An ACK packet to a CH
- A control request packet to a CH (to join a cluster)
- Data with critical information

If the data is being sent by a CH node, it could be

- An ACK to an ordinary sensor node
- Aggregated information to a sink node

The transmit data procedure is also used for the time duration where the UWSN client node is permitted to wait for the reception of an ACK packet. If the ACK packet is not forthcoming the client node can retransmit again to probe for a CH. This procedure also allows the client node to retry a number of times before it can be suspended into a sleep state. In fact, the number of retries is not limited but can be reset manually by the user at any time. Care should be taken to ensure that the time a sensor node waits for an ACK packet must be greater than the Round Trip Time (RTT) value (this value is calculated prior to sending the probe requesting membership to a cluster). The algorithm 4 illustrates the transmit data procedure. It details the type of packet received and the type of node the data is being delivered to. Again if the CH receives and forwards a request from a client node, then an ACK packet is transmitted otherwise the request control packet is discarded.

Step 1: Start

Step 2: Call check buffer procedure

Step 3: If (req_pkt received) then // request packet is the control packet

Step 4: If (Target_ID= =Sensor_ID) then // Sensor's Pkt's fields

Step 5: ACK successfully received

RESEARCH ARTICLE

Step 6: Else

Step 7: If (Target_ID = -1) then // if -1 then the sensor node is 'CH'

Step 8: Sensor is a CH - Send an ACK

Step 9: Else

Step 10: Discard data

Step 11: End If

Step 12: End if

Step 13: End If

Step 14: Stop

Algorithm 4 Transmit Data Procedure

5.2.6. Wait_For_Ack and Wait_For _Data Procedure

These two procedures are very essential since they define the time limits for the reception of an ACK packet and for data. The basic idea of clustering is to ensure that the member nodes communicate via the CH only. Sometimes the member nodes may receive irrelevant information or a data packet that is not intended for them. This situation may arise due to some of the following reasons listed below

- The destination node may receive duplicate packets
- The sensor node may have a received a packet intended for another node (the destination address is meant for another sensor node)
- A sensor node responsible for data collection may intentionally or accidentally listen to the data meant for other cluster members.

The Wait_For_Ack and Wait_For_Data procedures provide the necessary solution to the abovementioned issues encountered by the client sensor nodes. These procedures utilize the check buffer procedure routine to solve this problem. The check buffer procedure monitors the data received and cross checks the target node address .With the help of this information the Wait_For_Ack and Wait_For_Data procedures send the ACK packets to the appropriate sensor nodes and discards the irrelevant data packets. As for preventing the sensor nodes from listening to the data of the other nodes is beyond the scope of this manuscript. This issue will form the base of our future research work. The Wait_For_Ack procedure sets the timer limit for receiving the ACK packet while it also carries out the retry process for the client node to send out the probe packet for finding a CH. When the Client sensor node has not received an ACK packet, the procedure starts running and initiates the timer process (as shown in Algorithm 5).

Step 1: Start

Step 2: Set p_count // packet retry counter

Step 3: Iterate the process

Step 4: Switch on T2 // Timer

Step 5: Send data

Step 6: Do // Iterate the process

Step 7: If (There is no proper ACK) then

Step 8: Call Wait_For_Ack procedure

Step 9: End If

Step 10: If (T2 < RTT) then // RTT: Round Trip Time

Step 11: p_count=p_count+1

Step 12: End If

Step 13: While (p_count<limit)

Step 14: Stop

Algorithm 5 Wait_For_Ack Procedure

Algorithm 6 details the Wait_For _Data procedure which invokes the Check Buffer procedure to check its intended address and avoids the delivery of irrelevant packets to the client sensor nodes. Additionally, the Wait_For _Data Procedure also ensures that the CH in reception of a control request packet sends out the ACK within the stipulated time. As stated earlier, irrelevant packets are automatically discarded.

Step 1: Start

Step 2: Call Check_Buffer procedure

Step 3: If (req_pkt received) then // request packet is the control packet) then

Step 4: If (Target_ID = -1) then // if -1 then the sensor node is 'CH'

Step 5: Sensor is a CH - Send an ACK

Step 6: Else

Step 7: Discard data

Step 8: End If

Step 9: End If

Step 10: Stop

Algorithm 6 Wait_For _Data Procedure

5.2.7. Check_Buffer Procedure

This procedure is used to check the buffer of a sensor node to check whether it contains data and also to ascertain the data type. As mentioned before, this procedure works in tandem with the other two procedures (Check Buffer procedure) to ensure that the client sensor nodes do not receive unintended information at any time. It is also responsible for checking the buffer for storing the data that is received from the other sensor node (in the case of a CH). Algorithm 7 depicts the working of this procedure.

Step 1: Start

Step 2: Call Check_Buffer procedure

Step 3: If (Data_Pkt Received = = True) then // Data_Pkt - Data packet

Step 4: If (Data_Size !=0) then

Step 5: If (Target_ID= =Sensor_ID) then // Sensor's Pkt's fields

Step 6: Process to Store the data

Step 7: Send an ACK

Step 8: End If

Step 9: End If

Step 10: End If

Step 11: Stop

Algorithm 7 Check_Buffer Procedure

5.2.8. EEMCC Protocol Data Flow Diagram

The EEMCC routing protocol focuses on clustering the network and communicating the data collected to the intended destination. The EEMCC protocol consists of two main algorithms, namely,

- Clustering Algorithm
- Communication Algorithm

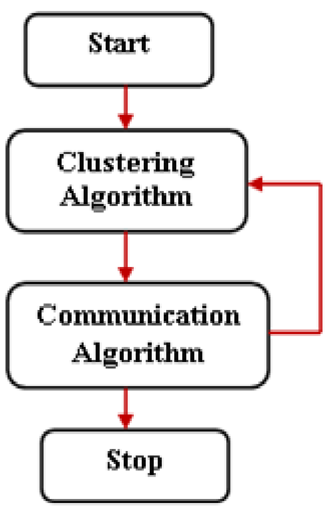


Figure 5 EEMCC Protocol Data Flow Diagram

The clustering algorithm is responsible for clustering the network and is in charge of forming the upper tier of the network. It also ensures that a suitable CH is selected along with a backup node. .It ensures that the clusters have suitable CHs and backup cluster nodes. The communication algorithm establishes a route between the wireless sensor nodes and the CH. Additionally, the communication algorithm ensures data collection and sends the collected data to the sink node or the selected CH nodes. Clustering usually involves dividing the network into smaller groups with a set of cluster members headed by a Cluster Head. This is expressed diagrammatically using data flow diagram as shown in Figure 5.

5.2.3. Phase II: Selection of Gateway Nodes (CH) by AUV for Communication

The AUV moves around the network communication range (region under study from which the data is collected) looking for suitable Cluster Heads to act as the gateway nodes .Initially, the AUVs move around the selected region while periodically broadcasting a Hello packet. This Hello packet is termed as a type1_broadcast message. The general format of this type of message is Broadcast_msg {type, start_time, End _Time}. The type field in this broadcast message indicates the node type, namely AUV or a CH (gateway node). The start_time and the end_ time fields indicate the time taken by the AUV to start travelling from an initial position S and its return to the same position (i.e. S).

When the broadcast message is received by a probable RGWN (Regional GateWay Node) sensor node, it computes the RSSI (received Signal Strength Indicator) value of the Hello packet. In order to select a suitable CH Gateway node, each sensor node broadcasts a type2 Broadcast_msg (a Hello control packet) with its Calculated RSSI value. The format of the type2 Broadcast_msg is given as Broadcast_msg {type, src_addr, RSSI value}, where the type field indicates the sensor node type .src_addr field represents the sensor node address and the RSSI field stores its corresponding value. When the type2 broadcast msg is received by the other nodes, they compare their RSSI value with the value stored in the type2 broadcast msg. the sensor node that has the highest RSSI value is chosen as the gateway node. All nodes in the network that transmit their collected data to the gateway nodes nearest to them. When the AUVs enter the transmission range of these gateway nodes, the Gateway nodes forward the collected data to the AUVs.

Residual energy thus plays a vital role in selecting the suitable gateway nodes. At periodic time intervals, the gateway nodes check their own residual energy levels. A threshold value is determined against which the gateway nodes check their RSSI values. The AUVs entering the region follow an elliptical trajectory path .the AUVS that are nearest to a gateway node gets the data from the nodes and

RESEARCH ARTICLE

transfers it to a communication link wither on the water surface or to the nearest base station if available. Our proposed EEMCCP ensures that the best AUV trajectory is selected in the deployment region. The trajectory of an AUV is specified by the following expression.

$$\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1 \quad (7)$$

Where

'a' - radii along x-axis

'b' – radii along y-axis

and $a^2 > b^2$

(h,k) - center of the ellipse

(h+a, k) and (h-a, k) are vertices end points of the minor axis.

5.2.3.1. Mathematical Notation for Obtaining a Set of Measured RSSI Values

The starting point of an AUV node is taken as x_1 and ends at x_m. The AUV moves along a trajectory path that is elliptical in nature. Throughout its travel, the AUV moves at a uniform speed. The AUV also uses a constant transmission power to transmit its location information. The trajectory is measured a t a length L. The distance between two neighboring points is denoted by ΔL. The total number of broadcasting points along the elliptical trajectory is denoted by equation (8).

$$m = {}^{L}/_{\Delta L} + 1 \quad (8)$$

While the AUV moves along its designated path, the UWSNs are responsible for transmitting the RSSI value throughout the network. The RSSI value thus measured and transmitted is depicted in equation (9).

$$\hat{r}\,(l) = SL - TL(l) + \tau \quad (9)$$

Where,

SL - Source Level

τ - random noise (variance - $\sigma^{2)}$

TL(l) - loss in transmission for an acoustic signal under water TL(l) is denoted in equation (10)

$$TL(l) = 10 * k * \log(l) + \propto . l \quad (10)$$

Where,

K – the spreading coefficient

α – absorption coefficient

l – distance covered by a transmission

The value of ' l ' is expressed in equation (11)

$$l = SQRT\ (x - x_N)^2 + (y - y_N)^2 + (z - z_N)^2 \quad (11)$$

Where

x, y, z - any point in the trajectory

x_n, y_n, z_{n-} sensor node location

As we know, the path of trajectory on the sea surface is a straight line, the value of z =0 and the point 'y' is linearly represented by 'x'.

The 'x' coordinate represents the broadcast points' location. The x- coordinates of all broadcast points along the path of trajectory join to form a location vector as represented in equation (12).

$$X = [x_{1,} x_{2,} x_{3} \ldots\ldots\ldots\ldots x_m] \quad (12)$$

Thus the RSSI values of all nodes can form a vector (equation (13))

$$\hat{R} = [\hat{r}_1, \hat{r}_{2,} \ldots\ldots\ldots, \hat{r}_m] \quad (13)$$

When the value of τ (equation (2)) equals 0, a noiseless RSSI value vector is attained (equation (14))

$$R = [r_1, r_2, \ldots \ldots \ldots \ldots ., r_m] \quad (14)$$

The UWSN nodes utilize the values of X and R_e. these values are used to determine the path of a UWSN on a trajectory and also to measure the perpendicular distance from a node to the trajectory. These values are used by the UWSN to learn about its location .The Algorithm 8 explains how the gateway nodes are selected by the EEMCCP.

As stated earlier, the Hello packet is broadcast .Any node receiving the Hello packet compares its RSSI value with the packet. If the node's RSSI value is not higher than the broadcast packet's RSSI value, it is passed on to the other CH nodes .The node having the maximum RSSI value is chosen to act as a gateway node .This gateway node receives the data from the cluster members, aggregates them and passes it on to the AUV that is nearer to it on the trajectory path.

Step 1: Start

Step 2: Set start time T1 and end Time T2

Step 3: While T1 < T2

Step 4: Broadcast Hello Packet

Step 5: End Loop (while)

Step 6: While true

Step 7: Node $_i$ receives HELLO packet (in one loop)

Step 8: Node $_i$ Broadcast RSSI

Step 9: If (RSSI (Node $_i$) < Z_{total} (RSSI)

Step 10: $Z_{GateWayNode}$ = Node $_i$

Step 11: Else

RESEARCH ARTICLE

Step 12: Step 7

Step 13: End

Step 14: Stop

Algorithm 8 Selection of Gateway Nodes (CH) by AUV

Figure 6 elaborates how our proposed scheme works. When the AUVs acquire the data from the gateway node they promptly pass it on to their nearest communication link. This communication link is responsible for transferring the data to a node on the water surface (say, a ship) or to the nearest terrestrial base station .The base station can then use its communication links to forward the data to the destination.

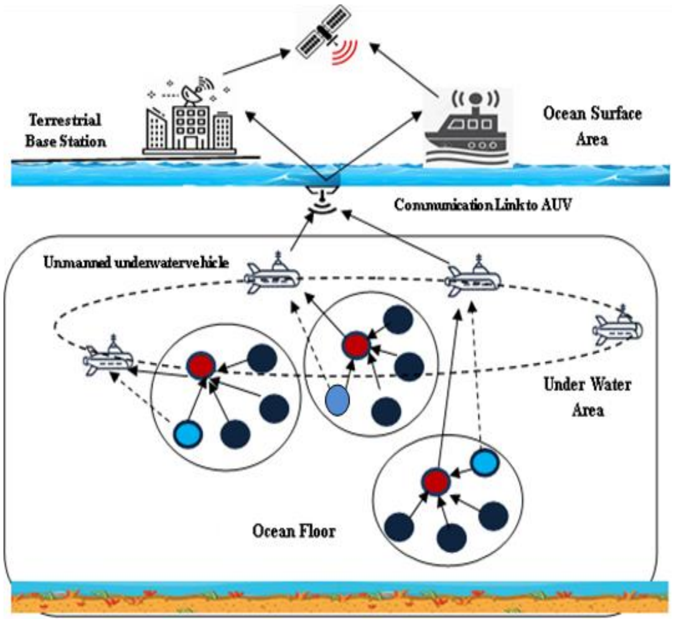


Figure 6 Selection of Gateway Nodes (CH) in the Clusters by AUV

6. RESULTS AND DISCUSSION

The proposed EEMCCP technique is compared against existing techniques like AEERP (AUV aided Energy Efficient Routing Protocol) and EEDBR (Energy Efficient Depth Based Routing) protocol. Both these protocols are applicable for underwater communication. This comparison helps us to analyze the performance of our proposed protocol over the existing ones. The comparison is carried out with the help of the Aqua Sim for NS2 simulator. The comparison is carried out based on major network metrics like:

- Energy consumption
- AUV's data collection
- PDR
- End – to –End Delay

6.1. AUV aided Energy Efficient Routing Protocol (AEERP)

The AEERP protocol is applied based on assumptions like

- Low network density
- Unlimited battery power
- The location of the nodes are always known

6.2. Energy Efficient Depth Based Routing (EEDBR) Protocol

This protocol has the following salient characteristics, namely,

- The sensor nodes are permitted to transmit based on their priority
- Works in two phase –knowledge acquisition phase and data transmission phase
- Priority value calculation for the nodes

6.3. Energy Consumption

The graph in Figure 7 shows the amount of energy consumed by the three protocols under study. The EEMCCP protocol shows a lesser consumption of energy than AEERP and the EEDBR protocol. The network energy is usually consumed when the nodes are made to do a lot of work with no reference to time. Again, the AEERP technique has a better energy consumption value than the EEDBR protocol. The AEERP is an efficient protocol, but, the overall framework and working is designed only after a set of fixed assumptions (high energy nodes, low network density and unlimited battery power). This may constitute for its better energy consumption values over EEDBR. The AEERP scheme does not support clustering as part of its design. The EEDBR protocol works on a different plane wherein, the priority of the nodes is calculated before the start of a transmission. Any node sending data has to suppress its energy if a high priority node wishes to transmit. It also works in two phases, where, in the first phase it acquires node data and then starts transmission in its second phase .The nodes spend time in sending information about themselves before transmission. This significantly reduces the battery power of the nodes. During the first phase the EEDBR protocol also carries out the task of calculating the priority of the nodes. All of these factors play a major role in reducing the battery power of the nodes. The proposed EEMCCP scheme first clusters the network and appoints a CH in addition to a backup cluster head. This allows efficient management of the network and also allots the power to a single CH rather than to all the nodes .The energy of the nodes is highly conserved due to this. The gateway node is made responsible for collecting the data from the sensor nodes. Again, the transfer of the data from the sensor nodes to the gateway node is stipulated by a timer. This helps to conserve the energy of the gateway node .This

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/195674 Volume 7, Issue 2, March – April (2020)

RESEARCH ARTICLE

helps the gateway node to conserve their energy (since it spends lesser time in collecting the data) efficiently pass on the data to the AUV nodes. Also, the RSSI value is used to determine the gateway node efficiently. The nodes that fail to join a cluster are suspended to save their energy. This is the reason why the energy consumption is lower in the proposed EEMCCP technique.

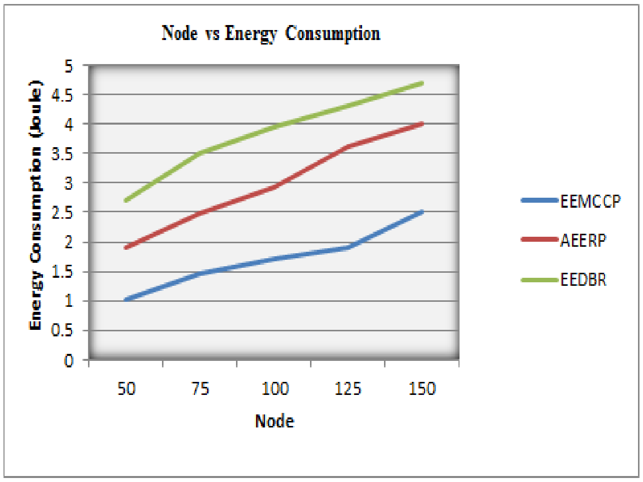


Figure 7 Energy Comparison

6.4. Total Amount of Data Collection by AUV

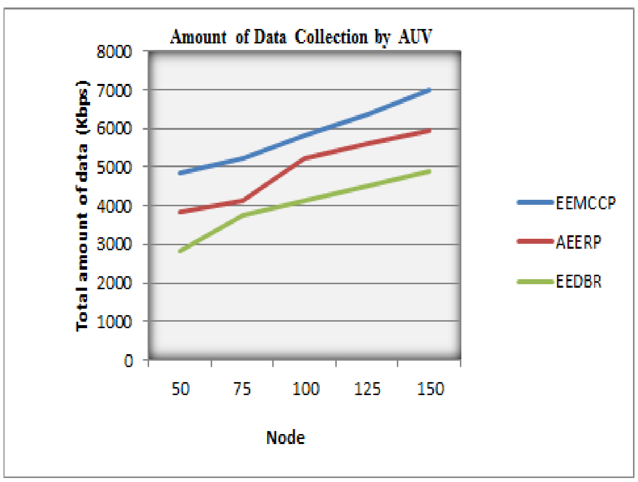


Figure 8 Amount of Data Collection by AUV

The total amount of data collected by the AUVs in the EEMCCP scheme than the other two protocols is shown in Figure 8. In fact, the graph shows an almost straight line indicating that the process of data collection is carried out smoothly than the other two methods. In fact, the graph for the AEERP and the EEDBR scheme shows uneven lines indicting their performance over the proposed method. Because of the clustering technique used in the EEMCCP scheme, the cluster operations are handled by the CH. The CH oversees the transfer of data (collected by the sensor nodes /cluster members) efficiently. Even if a CH fails, there is always a backup CH. Even the gateway nodes save time by transferring data to their nearest AUVs. The data collection is regulated by strictly adhering to time limits, thus efficiently managing the process. The AEERP scheme does not use clustering. This is a drawback since all the nodes act autonomously with no one to monitor them. In case there are inactive sensor nodes, it is not known till the AUV has arrived to collect the data. The EEDBR scheme has a priority calculation scheme for all the nodes. A low priority sensor node is usually suppressed from transmitting when a high priority node needs to transmit. This will naturally lead to a disruption in the data transfer phase .This reflects poorly on the data collection of the AUVs. The graph clearly shows the disruptions in the data collection phase.

6.5. Packet Delivery Ratio

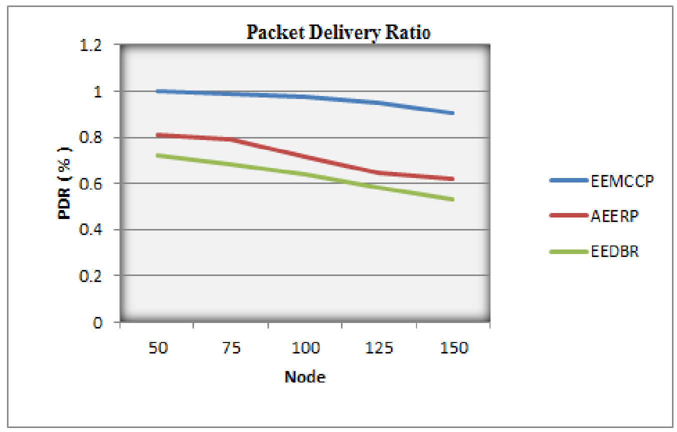


Figure 9 Packet Delivery Ratio Comparison

Figure 9 describes the PDR of the EEMCCP scheme is maximized than the AEERP and the EEDBR schemes. Naturally, as the node density increases, the PDR of the EEMCCP also decreases but only to a minimal extent. This can be attributed to the fact that the network is clustered with the CHs handling the transmission of data. Also when the CH of a cluster fails it is rectified without any delay with the help of the backup CH node .This helps in sustaining the flow of transmission without any delay. There is very minimal congestion as the CHs route the data to the gateways CHs. The AEERP scheme shows a minimal PDR when the network density increases. This may probably be due to the fact that the network is not clustered. All the sensor nodes are responsible for collecting and transferring the data to the gateway node, which may lead to a lot of congestion thereby disrupting the transmission. Again, the increase in the node density will lead to a reduced PDR. The AEERP scheme works based on a low network density scenario. Hence, the PDR of AEERP suffers greatly. The EEDBR scheme also suffers because of the fact that it does not cluster the network. When the node density increases, the PDR will naturally

RESEARCH ARTICLE

decrease since almost all the sensor nodes will be transmitting simultaneously. High priority nodes get to transmit first than the low priority nodes which may lead to starvation issues. This will definitely have a direct effect on the PDR of the network.

6.6. End-to-End Delay

The end-to-end delay refers to the time taken by a packet to travel from the source to the destination. It is evident from Figure 10 that the end-to-end-delay of the proposed EEMCCP is minimal than the other two methods. The sensor nodes that send a probe to join the cluster expect an ACK packet from a CH. IF a CH is not available then the requesting node is suspended after a number of stipulated tries. The EEMCCP does not waste time by allowing the nodes to continually broadcast, thus improving the data collection process. Also, the sensor nodes are bound by a timer to transfer data to the gateway nodes. The CHs take up the responsibility of transferring the data to the collected nodes, which in turn send it to the gateway nodes. The gateway nodes in turn ensure safe delivery of the data to the AUVs and thereon to the destination. All of this subsequently results in prompt delivery of the collected data to the destination. The lack of clusters in both the AEERP and EEDBR schemes leads to a lot of issues. Almost all the nodes are autonomous and any malicious nodes may induce unwanted chaos into the network. As stated earlier, the AEERP scheme works based on assumptions. If the node density increases, the AEERP scheme may behave erratically. This will affect the transmission of the data packet. It may get lost in the network traffic or may be lost altogether due to collision. The delay in the EEDBR scheme may result due to the priority based transmission scheme followed by the protocol. The high priority nodes may continually transmit while suppressing the low priority nodes. The low priority nodes may not have the chance to transmit or forward for longer periods. This may again result in duplicate packets (especially broadcasting) form the source thereby extending the delay in the network.

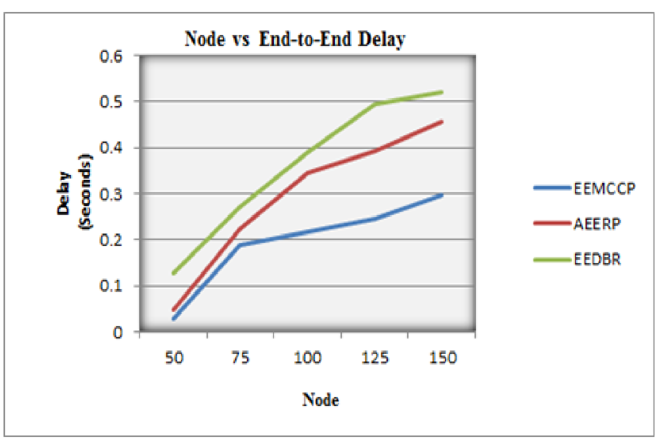


Figure 10 Delay Comparison

7. CONCLUSION

In this paper, we have proposed an Energy Efficient Minimum Cost Cluster routing Protocol (EEMCCP). This protocol plays a vital role in cluster based routing for UWSN. The EEMCCP first clusters the network and chooses a CH for each cluster. The scheme also selects a backup CH for every cluster .the cluster members do all their transactions via their respective CHs. Additionally, efficient CHs are selected based on their RSSI values to act as gateway nodes. This gateway transfers the collected data to the AUVs efficiently. The AUVs in turn send the data to their nearest communication point, from where the data is routed towards their destination. The overall working of all the sensor nodes are regulated efficiently by the protocol by using time controlled stipulations. This helps in conserving the energy of the nodes and also facilitates faster data transfer. The proposed protocol helps in maintaining a good PDR and also achieving a good throughput. The performance of the EEMCCP scheme is compared with two protocols, namely, AEERP and EEDBR. It is found that the EEMCCP scheme has a better performance statistics than the existing protocols .For further research; we have planned to focus on strengthening the security of the data packets and also to verify the trustworthiness of the transmitting nodes. This will help in further refining the EEMCCP technique and will result in better achievement of our objectives.

REFERENCES

[1] AG. A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh, and F. Hover, "Communication protocols for underwater data collection using a robotic sensor network," IEEE GLOBECOM Workshops, 2011, pp. 1308 –1313.

[2] Seokhoon Yoon, Abul K. Azad, Hoon Oh and Sunghwan Kim, "AURP: An AUV-Aided Underwater Routing Protocol for Underwater Acoustic Sensor Networks", Sensors 2012, 12(2), 1827 – 1845.

[3] F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges", Ad Hoc Networks, vol. 3, no. 3, pp. 257-279, 2005

[4] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Initialization and Routing Optimization for Ad-Hoc Underwater Acoustic Networks," Optical network – 2000

[5] Caiti A, Grythe K, Hovem JM, Jesus SM, Lie A, Munafo A, Reinen TA, Silva A, Zabel F. Linking acoustic communications and network performance: integration and experimentation of an underwater acoustic network. IEEE Journal of Oceanic Engineering. 2013 Oct; 38(4):758–71

[6] W. Zhang, M. Stojanovic, and U. Mitra, "Analysis of a simple Multihop underwater acoustic network," in Proceedings of the Third ACM International Workshop on Underwater Networks, ser. WuWNeT '08. New York, NY, USA: ACM, 2008, pp. 3–10.

[7] A. Stefanov and M. Stojanovic, "Design and performance analysis of underwater acoustic networks," IEEE Journal on Selected Areas in Communications, vol. 29, no. 10, pp. 2012–2021, December 2011.

[8] C. W. Ahn and J.-H. Lee, "Wireless cooperative communication: A survey," in Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication, ser. ICUIMC '11. New York, NY, USA: ACM, 2011, pp. 78:1–78:6.

[9] Dario Pompili and Tommaso Melodia, "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, ser. MobiCom '06. New York, NY, USA: ACM, 2006, pp. 298–309.

[10] Federico Favaro, Paolo Casari, Federico Guerra, Michele Zorzi, "Data Upload from a Static Underwater Network to an AUV: Polling or Random Access," IEEE Oceans 2012.

[11] D.A. Paley, F. Zhang, and N.E. Leonard, "Cooperative Control for Ocean Sampling: The Glider Coordinated Control System," IEEE Trans. Control Systems Technology, vol. 16, no. 4, pp. 735-744, July 2008.

[12] B. Liu, P.B.O. Dousse, P. Nain, and D. Towsley, "Mobility Improves Coverage of Sensor Networks," Proc. ACM MobiHoc, 2005

[13] Y. Yu and J.L. Rittle, "Utility-Driven Spatiotemporal Sampling in Mobile Sensor Networks," Proc. IEEE INFOCOM '08, Apr. 2008.

[14] Geoffrey A. Hollinger, Sunav Choudhary, Parastoo Qarabaqi, Christopher Murphy, Urbashi Mitra, Gaurav S. Sukhatme, Milica Stojanovic, Hanumant Singh, Franz Hover, "Autonomous Data Collection from Underwater Sensor Networks using Acoustic Communication," IEEE Journal on Selected Areas in Communication 2012.

[15] E. Balas, "The Prize Collecting Traveling Salesman Problem," Networks, vol. 19, pp. 621–636, 1989.

[16] H. Bojinov, A. Casal, and T. Hogg, "Emergent structures in modular self-reconfigurable robots," in Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1734–1741, San Francisco, Calif, USA, April 2000.

[17] AG. A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh, and F. Hover, "Communication protocols for underwater data collection using a robotic sensor network," IEEE GLOBECOM Workshops, 2011, Dec. 2011, pp. 1308 –1313.

[18] Sathiamoorthy, J., and B. Ramakrishnan. "Energy and delay efficient dynamic cluster formation using improved ant colony optimization algorithm in EAACK MANETs." *Wireless Personal Communications* 95, no. 2 (2017): 1531-1552.

[19] Harb H, Makhoul A, Couturier R. An enhanced K-means and ANOVA-based clustering approach for similarity aggregation in underwater wireless sensor network. IEEE Sensors Journal. 2015 Oct; 15(10):5483–93.

[20] Sathiamoorthy, J., and B. Ramakrishnan. "A competent three-tier fuzzy cluster algorithm for enhanced data transmission in cluster EAACK MANETs." *Soft Computing* 22, no. 19 (2018): 6545-6565.

[21] Sathiamoorthy, J., B. Ramakrishnan, and M. Usha. "A reliable and secure data transmission in CEAACK MANETs using distinct dynamic key with classified digital signature cryptographic algorithm." In *2015 International Conference on Computing and Communications Technologies (ICCCT)*, pp. 144-151. IEEE, 2015.

[22] Forghani, Azadeh, Amir Masoud Rahmani, and Ahmad Khademzadeh. "QCTC: QoS-based clustering topology control algorithm for wireless sensor networks." In *2008 International Conference on Advanced Computer Theory and Engineering*, pp. 966-970. IEEE, 2008.

[23] Sathiamoorthy, J., Ramakrishnan, B., & Usha, M. (2015). Design of a competent broadcast algorithm for reliable transmission in CEAACK MANETs. Journal of Network Communications and Emerging Technologies, 5(1), 144–151.

[24] Taheri, H., Neamatollahi, P., Younis, O. M., Naghibzadeh, S., & Yaghmaee, M. H. (2012). An energyaware distributed clustering protocol in wireless sensor networks using fuzzy logic. Journal of Ad Hoc Networks, 10(7), 1469–1481.

[25] Bagci, H., & Yazici, A. (2010). An energy aware fuzzy unequal clustering algorithm for wireless sensor networks. Applied Soft Computing, 13(4), 1741–1749.

[26] Pranay Kumar Baghel and Anil Kumar Mishra (2016), Comparative Analysis of Different Approaches for Navigation and Path Planning, International Journal of Emerging Technologies in Engineering Research (IJETER), Volume 4, Issue 6,2016

[27] Ravinder Kaurand Dr. Neeraj Sharma (2015), A Node Authentication Mechanism to Enhance the Security in VANETs, International Journal of Emerging Technologies in Engineering Research (IJETER), Volume 1, Issue 2,2015

[28] Er. Jagdeep Kaurand Prof. (Dr.) Jatinder Singh (2015), Particle Swarm Optimization Based Neural Network, International Journal of Emerging Technologies in Engineering Research (IJETER), Volume 3, Issue 2, 2015

[29] Linfeng Liu, Ye Liu," On Exploiting Signal Irregularity with Topology Control for Mobile Underwater Wireless Sensor Networks", proc. IEEE Globecom - Ad Hoc and Sensor Networking Symposium, pp. 522-527, 2013.

[30] Seokhoon Yoon, Abul K. Azad, Hoon Oh and Sunghwan Kim, "AURP: An AUV-Aided Underwater Routing Protocol for Underwater Acoustic Sensor Networks", Sensors 2012, 12(2), 1827 – 1845.

[31] Chen, Yuh-Shyan, and Yun-Wei Lin. "Mobicast routing protocol for underwater sensor networks." *IEEE Sensors journal* 13, no. 2 (2012): 737-749.

[32] Favaro, Federico, Paolo Casari, Federico Guerra, and Michele Zorzi. "Data upload from a static underwater network to an AUV: Polling or random access?." In 2012 Oceans-Yeosu, pp. 1-6. IEEE, 2012.

Authors

M. Usha is a research Scholar, currently pursuing her Ph.D from M.S.University, Tirunelveli, Tamil Nadu, India. She has completed M.C.A. and M.Phil in Computer Science from Bharathidasan University, Trichy. She has also done her M.Tech (CIT) from Manonmaniam Sundaranar University, Tirunelveli. She is currently working as an Associate Professor in MEASI Institute of Information Technology, Chennai. She has 12 years of teaching experience and she has published papers in Network Security in National and International journals and has presented in International Conferences and Seminars. Her current area of interest is Operating Systems, Algorithms and ad-hoc networks especially MANET, VANET, FANET and Underwater Communication.

Dr. J.Sathiamoorthy is currently working as an Associate Professor in Thiruthangal Nadar College, Chennai. He has completed M.C.A. and M.Phil in Computer Science from University of Madras and Vinayaka Mission University respectively. He has also done his M.Tech (CIT) and Ph.D from Manonmaniam Sundaranar University, Tirunelveli. He has published papers in Network Security in National and International journals and has presented in International Conferences and Seminars. His current area of interest is ad-hoc networks especially MANET, VANET, FANET and Underwater Communication.

R. Ashween is a final year BE Computer Science and Engineering student. He is pursuing his BE degree form Amrita School of Engineering, Coimbatore, Tamil Nadu, India. His research interests include Wireless Sensor Network and Mobile Ad Hoc Networks.

Bhagavath Nishanth Ramakrishnan has a degree in Electronics and Communication Engineering from Velammal Engineering College, affiliated to Anna University. He graduated recently with a Master of Information and Communication Technology degree from Western Sydney University, NSW, Australia. His research interests include Network Communication, Mobile Communication and Network Security especially MANET, VANET and Underwater Communication.