Bounded Side-based Clustering VBF Routing Protocol in Underwater Wireless Sensor Networks

Dina M. Ibrahim, Tarek E. Eltobely, Mahmoud M. Fahmy, and Elsayed A. Sallam
Computers and Control Engineering Department
Faculty of Engineering, Tanta University
Tanta, Egypt
emails: {dina.mahmoud@f-eng.tanta.edu.eg, tarekt@f-eng.tanta.edu.eg, mfn_288@hotmail.com, and sallam@f-eng.tanta.edu.eg}

***Abstract*— Underwater Wireless Sensor Networks (UWSNs) have an important role in different applications, such as offshore exploration and ocean monitoring. In this paper, we improve the performance of the Clustering Vector-Based Forwarding algorithm (CVBF) by enhancing its behavior. Here, the whole network is divided into a predefined number of clusters and determining an internal bounded side with its own coordinates for each cluster. The main target of our proposed algorithm is to avoid the calculations taken for the nodes in which they never move away from its cluster. These nodes' positions are near the cluster axis and far enough from the cluster boundaries. For this reason, we separate between the nodes that can move outside the clusters and the nodes that never move away. Simulation results demonstrate that the proposed algorithm reduces the energy consumption especially in dense networks, increases the packet delivery ratio especially in sparse networks, and decreases the average end-to-end delay in both sparse and dense networks. These advantages are emphasized when the algorithm is compared with five other powerful routing algorithms: VBF, Hop-by-Hop VBF (HH-VBF), Vector-Based Void Avoidance (VBVA), Energy-Saving VBF (ES-VBF), and the CVBF routing protocols.**

Keywords-wireless networks; underwater sensor networks; multiple clusters; routing protocols; CVBF; BS-CVBF.

I. INTRODUCTION

At the end of the twentieth century, wireless sensor networks became a hot research area. At the beginning, these networks covered only terrestrial applications. However, the earth is known to be a water planet, with 70 % of its surface being covered with water (principally oceans). With the increasing role of oceans in human life, discovering all of the ocean parts became of prime importance. On one side, traditional approaches formerly used for underwater monitoring missions have several drawbacks and on the other side, these harsh environments are not feasible for human presence as unpredictable underwater activities, high water pressure, predatory fish and vast areas are major reasons for un-manned exploration. Due to these reasons, Underwater Wireless Sensor Networks (UWSNs) attract the interest of many researchers lately [1]. Over the last three decades, significant contribution has been made in the area of scientific, commercial, and military applications [2][3][4][5]. In particular, highly precise real-time continuous-monitoring systems are essential for vital operations such as off-shore oil field monitoring, pollution detection, disaster prevention, assisted navigation, mine reconnaissance, and oceanographic data collection [6]. All these significant applications call for building UWSNs. The work done by Akylidiz et al. [7] is considered as the pioneering effort towards the deployment of sensor nodes for underwater environments.

Though there exist many network protocols for terrestrial wireless sensor networks, the underwater acoustic communication channel has its unique characteristics, such as limited bandwidth capacity and high delays, which require new efficient and reliable data communication protocols [8][9][10]. Major challenges in the design of underwater wireless sensor networks are: i) the limited bandwidth; ii) the underwater channel is severely impaired, especially due to multipath and fading problems; iii) high propagation delay in underwater which is five orders of magnitude higher than in Radio Frequency (RF) terrestrial channels; iv) high energy consumption due to longer distances; v) battery power is limited and usually batteries cannot be recharged, also because solar energy cannot be exploited underwater; vi) underwater sensor nodes are prone to failures due to fouling and corrosion. All the factors mentioned above, especially limited energy, would make designing a routing protocol for UWSN an enormous challenge.

Routing is a fundamental issue for any network, and routing protocols are considered to be in charge of discovering and maintaining the routes [6]. Most of the research works concerning UWSNs have been on the issues related to the physical layer, while issues related to the network layer such as routing techniques are a relatively new area. Thus, an efficient routing algorithm is to be provided. Although underwater acoustic networks have been studied for decades, underwater networking and routing protocols are still at the infant stage of research.

A review of underwater network protocols till the year 2000 can be found in [2]. Several routing protocols have been proposed for underwater sensor networks. A good survey until year 2012 about underwater wireless sensor routing techniques is presented in [6]. Here, Ayaz et al. introduced an overview of the state of the art of routing protocols in UWSNs and thoroughly highlighted the advantages, functionalities, weaknesses and performance issues for each technique. Based on network architecture, UWSNs routing protocols are classified into: location-based, flat, and hierarchical routing protocols. Vector-Based Forwarding (VBF) protocol has been suggested in order to solve the problem of high error probability in dense networks [11]. It is a location-based routing protocol. Here an idea of a virtual routing pipe from the source to the destination is proposed, and all the flooding data packets are carried out through this pipe. An enhanced version of VBF called Hop-by-Hop VBF (HH-VBF) has been proposed [12]. They use the same concept of virtual routing pipe as used by VBF, but instead of using a single pipe from source to destination, HH-VBF defines per hop virtual pipe for each forwarder [13]. Another extension of VBF protocol is introduced in [14] called Vector-Based Void Avoidance (VBVA) routing protocol, which extends the VBF routing protocol. It addresses the routing void problem in underwater sensor networks. VBVA assumes two mechanisms, vector-shift and back-pressure, to handle voids. In [15], an energy-aware routing algorithm, called Energy-Saving Vector Based Protocol (ES-VBF), is proposed. In this protocol, Bo et al. put forward an energy-aware routing algorithm to save network energy. It takes both residual energy and location information into consideration, which shows a promising performance in balancing network energy consumption and packet reception ratio.

In our recent research [1], we propose a Clustering Vector-Based Forwarding algorithm (CVBF) in order to improve the performance of the VBF protocol. In the proposed algorithm, the network space volume is divided into a number of clusters where one virtual sink is assigned to each cluster. Inside each cluster, all the nodes are allowed to communicate with themselves just to reach its virtual sink node, which in turn sends the packets to the main sink in the network. Simulation results demonstrate that the proposed algorithm reduces the energy consumption especially in dense networks, increases the packet delivery ratio especially in sparse networks, and decreases the average end-to-end delay in both sparse and dense networks. These advantages are emphasized when the algorithm is compared with five other powerful routing algorithms: VBF, HH-VBF, VBVA, ES-VBF, and CVBF routing protocols.

Other UWSNs routing protocols, such as Dynamic Source Routing (DSR), Focused beam Routing (FBR), Directional Flooding-Based (DFR), and Depth-Based Routing (DBR) are found in [6][13][16][17].

The remainder of this paper is organized as follows. In Section II, the behavior and performance issues of VBF, HH-VBF, VBVA, ES-VBF, and CVBF location-based routing protocols, which will be used in a comparison with our algorithm, are discussed. Section III presents the details of the proposed algorithm. In Section IV, the performance under varying the number of clusters are discussed. Then, we show the performance results of the proposed algorithm in Section V. Finally, we draw the main conclusions in Section VI.

II. LOCATION-BASED ROUTING PROTOCOLS: A REVIEW

In this section, we briefly discuss the five location-based routing protocols, which we will select to compare our algorithm with. These protocols are presented in the following:

A. *Vector-Based Forwarding (VBF) Routing Protocol*

VBF is a location-based routing approach for UWSNs proposed by Xie et al. [11]. In this protocol, state information of the sensor nodes is not required since only a small number of nodes are involved during packet forwarding. Data packets are forwarded along redundant and interleaved paths from the source to the sink, which helps handling the problem of packet losses and node failures. It is assumed that every node previously knows its location, and each packet carries the location of all the nodes involved including the source, forwarding nodes, and final destination. The forwarding path is specified by the routing vector from the sender to the target. As soon as a packet is received, the node computes its relative position with respect to the forwarder. Recursively, all the nodes receiving the packet compute their positions. If a node determines that it is close enough to the routing vector, it puts its own computed position in the packet and continues forwarding the packet; else, it simply discards the packet. In this way, all the packet forwarders in the sensor network form a "routing pipe", the sensor nodes in this pipe are eligible for packet forwarding, and those that are not close to the routing vector do not forward. Fig. 1 illustrates the basic idea of VBF. In this figure, node S_1 is the source, and node S_0 is the sink. The routing vector is specified by S_1S_0. Data packets are forwarded from S_1 to S_0. Forwarders along the routing vector form a routing pipe with a pre-controlled radius, W.

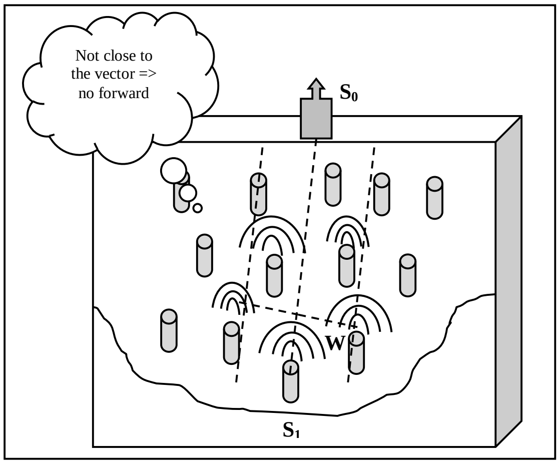


Figure 1. VBF routing protocol for UWSNs.

Additionally, a localized and distributed self-adaptation algorithm is developed to enhance the performance of VBF [11]. The self-adaptation algorithm allows each node to estimate the density in its neighborhood and forward packets adaptively. This algorithm is based on the definition of a desirableness factor, α [11]. This factor measures the suitability of a node to forward packets. Given a routing vector S_1S_0 and forwarder F, the desirableness factor of a node A is:

$$\propto = \frac{P}{W} + \frac{R - d\times\cos\theta}{R} \tag{1}$$

where P is the projection length of A onto the routing vector S_1S_0, d is the distance between node A and node F, θ is the angle between vector FS_0 and vector FA, R is the transmission range, and W is the radius of the routing pipe.

VBF has many essential drawbacks. First, using a virtual routing pipe from source to destination can affect the routing efficiency of the network with different node densities. In some spaces, if node deployment is sparser or becomes sparse due to some node movement, then it is possible that very few or even no node will lie within that virtual pipe, which is responsible for the data forwarding; even it is possible that some paths may exist outside the pipe. Eventually, this will result in small data deliveries in sparse spaces. Second, VBF is very sensitive about the routing pipe radius threshold, and this threshold can affect the routing performance significantly; such feature may not be desirable in the real protocol developments. Furthermore, some nodes along the routing pipe are used again and again in order to forward the data packets from sources to the sink, which can exhaust their battery power.

B. HH-VBF Routing Protocol

The need to overcome two problems encountered by the VBF, i.e., small data delivery ratio in sparse networks, and sensitivity to the routing pipe's radius, the HH-VBF (hop-by-hop VBF) is proposed by Nicolaou et al. [12]. HH-VBF forms the routing pipe in a hop-by-hop method, enhancing the packet delivery ratio significantly. Although it is based on the same concept of routing vector as VBF, instead of using a single virtual pipe from the source to the sink, it defines a different virtual pipe around the per-hop vector from each forwarder to the sink. In this protocol, each node can adaptively make packet forwarding decisions based on its current location. This design can directly bring the following two benefits: First, since each node has its own routing pipe, the maximum pipe radius is the transmission range. Second, in sparse networks, HH-VBF can find a data delivery path even so the number of eligible nodes may be small, as long as there exists one in the network.

In HH-VBF, the routing virtual pipe is redefined to be a per-hop virtual pipe, instead of a unique pipe from the source to the sink [12]. When some areas of the network are not occupied with nodes, for example, there exist "voids" in the network, even a self-adaptation algorithm may not be able to route the packets. In such a case, a forwarder is unable to reach any node other than the previous hop. Although simulation results show that HH-VBF considerably produces better results for packet delivery ratio, but still it has an inherent problem of routing pipe radius threshold, which can affect its performance. Moreover, due to its hop-by-hop nature, HH-VBF is not able to add a feedback mechanism to detect and avoid voids in the network and energy efficiency is still low compared to VBF [12].

C. VBVA Routing Protocol

Xie et al. [14] introduce a Vector-Based Void Avoidance (VBVA) routing protocol, which extends the VBF routing protocol to handle the routing void problem in UWSNs. VBVA assumes two mechanisms, vector-shift and back-pressure. The vector-shift mechanism is used to route data packets along the boundary of a void. The back-pressure mechanism routes data packets backward to bypass a concave void. VBVA handles the routing void problem on demand and thus does not need to know network topology and void information in advance. Hence, it is very robust to cope with mobile voids in mobile networks. Simulation results in [14] show that VBVA can handle both concave and convex voids effectively and efficiently in mobile underwater sensor networks only when these voids are inside the forwarding pipe, while the voids outside the forwarding pipe is not solved by VBVA.

D. ES-VBF Routing Protocol

To solve the energy problem in UWSN, Bo et al. [15] put forward an energy-aware routing algorithm, called Energy-Saving Vector-Based Protocol (ES-VBF). The main purpose of this routing protocol is saving energy. ES-VBF takes both residual energy and localization-based information into consideration while calculating the desirableness factor as in (2), which allows nodes to weigh the benefit for forwarding packets. The ES-VBF algorithm modifies the calculation of the desirableness factor of (1) for VBF protocol to be calculated if the node residual energy is smaller than 60% of initial energy as:

$$\propto = 0.5 \times \left(1 - \frac{energy}{initialenergy}\right) + \left(\frac{P}{W}\right) + \left(\frac{R - d \times \cos\theta}{R}\right) \quad (2)$$

where *energy* is the residual energy of nodes and *initialenergy* is the initial energy of nodes. By simulation results in [15], it is shown that the performance is promising in balancing network energy consumption and packet reception ratio. This means that the ES-VBF protocol saves energy in an efficient manner. At the same time, there is a small falling in packet reception ratio, which needs further research aiming at finding a better solution not only reducing energy consumption but also achieving high packet reception ratio.

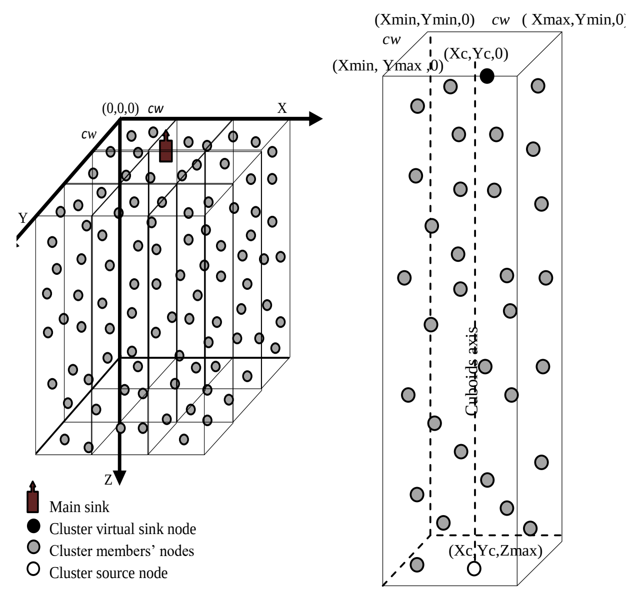
E. Clustering VBF (CVBF)Routing Protocol

In this algorithm (our recent research) the authors propose an algorithm for UWSNs called CVBF algorithm. The objective of the CVBF routing algorithm is to reduce energy consumption, increase the packet delivery ratio, and decrease the average end-to-end delay.

In [1], the whole network is divided into a predefined number of clusters. All sensor nodes are assigned to the clusters on the basis of their geographic location, and then one node at the top of each cluster is selected as a *virtual sink* for that cluster. The rest of nodes in each cluster transmit the data packets to their respective cluster virtual sink. The routing inside each cluster follows the VBF routing protocol discussed in Section II. This implies that the concept of using one virtual routing pipe for all network nodes in VBF is replaced by defining one virtual routing pipe for each cluster to forward the packets from any node in the cluster to its virtual sink in that cluster. We assume that the routing pipe radius is equal to the transmission range of a node. Each intermediate node in any cluster selects the next hop to a node inside its cluster. In this way, the network will have many virtual routing pipes, one pipe per cluster, which guarantees forwarding the packets in the upward direction instead of forwarding the packets widely across the network nodes in the VBF algorithm. It is well expected that this will decrease the average end-to-end delay node and reduce the number of hops to reach the virtual sink node, which will enhance the network performance. In addition, CVBF avoid voids in the network because each node belongs to a specific cluster.

Also, if a small number of nodes are available in the neighborhood, CVBF can still find a data delivery path. After receiving the data packets from cluster sensor nodes, cluster virtual sinks perform an aggregation function on the received data, and transmit them towards the main sink node using single-hop routing. Cluster virtual sink nodes are responsible for coordinating their cluster members and communicating with the main sink node. The proposed algorithm consists of 3 steps: Step1, which called *clustering the nodes*, step 2 is *selecting the cluster virtual sink*, and step 3, which called *calculating the cluster's maintenance time*.

Step 1 is responsible for dividing the network into groups of nodes according to their geographic location producing non-overlapping clusters excluding the main network sink, which is allocated on the water surface. The following values are given: the network space X×Y×Z, node transmission range, routing pipe width, and the node speed. We divide the network space into equal space volumes; in the form of cuboids [1]. The division is based on the values of X and Y coordinates, and the cluster width, *cw*, as shown in Fig. 2 (a).



(a) Network area with 9 clusters (b) One cluster and its virtual sink

Figure 2. A CVBF network area: (a) Network area with 9 clusters, (b) One cluster and its virtual sink

Choosing the best number of clusters is proposed as:

$$N = \frac{X \times Y}{(cw)^2} \tag{3}$$

where $X \times Y$ is the total surface area of the network and $(cw)^2$ is the area of the cluster surface. The cluster width is thus calculated as:

$$cw = \sqrt{\left(\frac{X \times Y}{N}\right)} \tag{4}$$

It is given that the surface area is square; therefore, we choose N as a number raised to the power of two: 2^2, 3^2, 4^2, or 5^2. Here, we choose N that gives the value of *cw* as near as possible to $\sqrt{2}$R in order to make sure that the virtual pipe of the cluster includes all the nodes inside that cluster.

Step 2 selects the cluster virtual sink for each cluster (cuboid). Each cluster has a space volume of cw×cw×Z, we choose the nearest node to the main sink to be a cluster virtual sink. As shown in Fig. 2 (b), the surface corner coordinates of the cluster are: (Xmin, Ymin, 0), (Xmax, Ymin, 0), (Xmin, Ymax, 0), and (Xmax, Ymax, 0).

All other nodes can send data to their corresponding virtual sink following the mechanism of VBF and depending on the value of its desirableness factor α. If more than one node has the same depth position, we choose the nearest node to the cuboid axis, in which its surface point coordinates is the point (Xc, Yc, 0). The source node of the cluster is fixed at the position (Xc, Yc, Zmax).

Step 3 takes into consideration the node mobility that affects network topology and performance, thus necessitating a cluster maintenance algorithm. For a correct network operation, the maintenance algorithm should be executed simultaneously in all clusters. In this step, we propose a suitable periodical time, which we call *maintenance time, Tm*. This time is enough to move a node from its cluster to another cluster according to speed and maximum distance of the node. Each node in the cluster checks its belonging to that cluster after the periodical time *Tm*. If a node belonging to a cluster moves away from that cluster, it naturally has two choices. The first choice is to enter another neighboring cluster, and so we transfer this node from the old cluster to the new cluster. The second choice is that it exits from all the network space, and so we leave this node in the old cluster.

To calculate *Tm*, we divide the known maximum distance of a node movement, *dmax*, by the current speed of the node, *S*:

$$Tm = \frac{dmax}{S} \tag{5}$$

In other words, all the nodes with positions near the cluster boundaries are prone to exit from their own cluster and enter to other clusters. One of the CVBF algorithm drawbacks is the calculations of *Tm* for all the nodes that take much time while there is a wasted time calculated with the nodes resides on the middle of each cluster, which never have a chance to move away from the cluster.

III. BOUNDED SIDE-BASED CLUSTERING VBF: THE PROPOSED ALGORITHM

In this section, we propose an algorithm for the routing protocols in UWSNs, which we call a Bounded Side-based Clustering VBF (BS-CVBF), which is an extended version of our recent paper in [1]. The main objective of this algorithm is to enhance the performance of the clustering VBF. This enhancement includes the energy consumption, the packet delivery ratio (PDR), the average end-to-end delay, and the CPU utilization. These four metrics are emphasized through comparison with five location-based routing protocols; VBF, HH-VBF, VBVA, ES-VBF, and CVBF protocols.

According to our algorithm, the whole network is divided into a predefined number of clusters follows step 1 in the CVBF routing algorithm discussed in Section II. In this way, we add another substep to step 1, which is responsible for determining internal bounded side with its own coordinates for each cluster, as shown in Fig. 3. The main target of our proposed algorithm is to avoid the calculations taken for the nodes in which they never move away from its cluster. These nodes' positions are near the cluster axis and far enough from the cluster boundaries. For this reason, we separate between the nodes that can move outside the clusters and the nodes that never move away.

This action is controlled by dividing the volume area of each cluster, cuboid, into two volume areas: The first volume area, which we call the inner cuboid, bounded by the (Xsmin, Ysmin, 0), (Xsmax, Ysmin, 0), (Xsmin, Ysmax, 0), and (Xsmax, Ysmax, 0). The second volume area is the rest of the cluster volume area, which represented by a gray color in Fig. 4.

The proposed algorithm is stated in the following steps:

Step1: Clustering the Nodes and Creating the Bounded Side

In addition to step 1 in the CVBF [1] we divide each cluster into two nested cuboids, as presented in Fig. 4. The inner cuboid contains the number of nodes that never move away from the cluster at any time during the network life time. The reason is our choice to the inner cuboid boundaries is calculated based on the maximum distance of a node movement where:

$$Xsmin = Xmin + dmax \tag{6}$$

$$Xsmax = Xmax - dmax \tag{7}$$

$$Ysmin = Ymin + dmax \tag{8}$$

$$Ysmax = Ymax - dmax \tag{9}$$

Step2: Selecting the Cluster Virtual Sink

For each cluster that has a space volume cw×cw×Z, we choose the nearest node to the main sink to be a cluster virtual sink. Like the CVBF algorithm. As shown in Fig. 2 (b), the surface corner coordinates of the cluster are: (Xmin, Ymin, 0), (Xmax, Ymin, 0), (Xmin, Ymax, 0), and (Xmax, Ymax, 0).

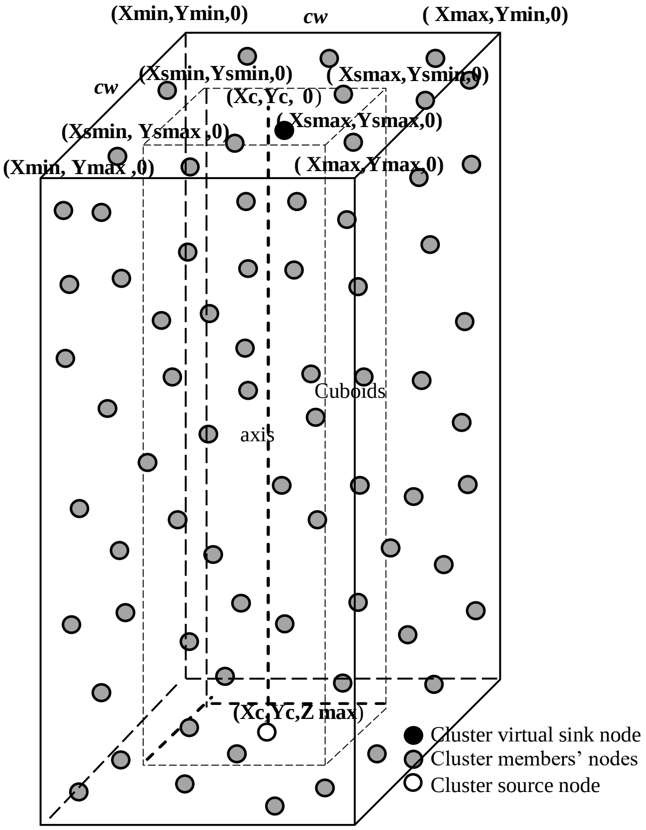


Figure 3. One cluster with the inner bounded side.

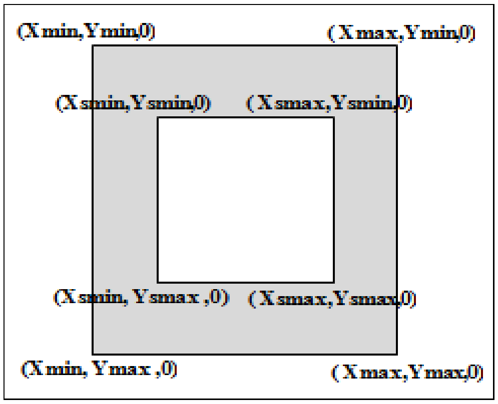


Figure 4. A horizontal section of a cluster including the inner bounded side.

All the other nodes can send data to their corresponding virtual sink following the mechanism of VBF and depending on the value of its desirableness factor α. If more than one node has the same depth position, we choose the nearest node to the cuboid axis, in which its surface point coordinates is the point (Xc, Yc, 0). The source node of the cluster is fixed at the position (Xc, Yc, Zmax).

Step3: Calculating the Cluster's Maintenance Time with respcet to the bounded side

This step takes into consideration the node mobility that affects network topology and performance, thus necessitating a cluster maintenance algorithm. For a correct network operation, the maintenance algorithm should be executed simultaneously in all clusters. In this step, we propose a suitable periodical time which we call *maintenance time, Tm.* This time is enough to move a node from its cluster to another cluster according to speed and maximum distance of the node. All the nodes inside the bounded side, represented by the gray color in Fig. 4 checks its belonging to that cluster after the periodical time *Tm*. If a node belonging to a cluster moves away from that cluster, it naturally has two choices. The first choice is to enter another neighboring cluster, and so we transfer this node from the old cluster to the new cluster. The second choice is that it exits from all the network space, and so we leave this node in the old cluster.

The proposed algorithm is summarized in the Pseudocode of Fig. 5.

```
Pseudocode
Step 1: Clustering the nodes and creating the bounded side
  1.1. All the substeps in the CVBF algorithm
  1.2. Determining the coordinates of the inner cuboid as:
         (a)   $Xsmin = Xmin + dmax$
         (b)   $Xsmax = Xmax - dmax$
         (c)   $Ysmin = Ymin + dmax$
         (d)   $Ysmax = Ymax - dmax$

Step 2: Selecting the cluster virtual sink
  1. All the substeps in the CVBF algorithm
Step 3: Calculating the cluster's maintenance time with
    respcet to the bounded side
  1. Any node is belonging to the inner cuboid (has Xis
     and Yis coordinates)
     if
        Ximin>=Xis>=Xsmin  AND
        Yimin>=Yis>=Yimax
     OR
        Xsmax>=Xis>=Ximax  AND
        Yimin>=Yis>=Yimin
     OR
        Xsmin>=Xis>=Xsmax  AND
          Yimin>=Yis>=Ysmax
     OR
          Xsmin>=Xis>=Xsmax  AND
          Ysmax>=Yis>=Yimax
     Then
        This node may move away from it cluster
     Else
        If a node exits from the cluster
        Then
           (a)Given the node speed, S, and the maximum
              distance of any node, dmax,
           (b) Calculate Tm=dmax/S
           (c) For J=0 to Simulation time with step Tm
                 For each node in the cluster i and has
                 coordinates (Xi,Yi,Z)
                     If (Xsmin>=Xi>=Xsmax and
                        Ysmin>=Yi>=Ysmax)
                     Then
                        This node is still in the cluster
                     Else
                        Remove this node from cluster i
                        and enter it to the suitable
                        neighboring cluster
  2. All the nodes in cluster i forward the packets to
     its virtual sink following the mechanism of VBF
     routing algorithm
  3. All the virtual sinks forward the packets to the
     main sink
```

Figure 5. Pseudocode of the proposed routing protocol BS-CVBF.

IV. PERFORMANCE EVALUATION

Performance is quantified through measures of energy consumption, packet delivery ratio, average end-to-end delay, and the CPU utilization percentage [15]. The energy consumption is the total energy consumed by the sensor network nodes. The packet delivery ratio is the rate of the number of packets successfully received by the sink to the number of packets generated by the source. The average delay is the average end-to-end delay for each packet received by the sink.

Simulation is performed by the underwater package Aqua-Sim of ns-2 [18][19] version ns2.35. In all our simulations, we set the parameters similar to UWM1000 LinkQuest Underwater Acoustic Modem [20]. All the network parameters are described in Table I.

TABLE I. NETWORK PARAMETERS

Parameter	***Value***
Network Space Volume	600m × 600m ×600m
Data packet size	76 bytes
Bt rate	10 kbps
Sending Energy	0.6 J
Receiving Energy	0.3 J
Idle Mode Energy	0.01 J
Node Speed	2m/s – 5m/s
Number of clusters	9 clusters
Total simulation time	1000 S
Number of the simulator running	30

The simulation results are plotted in Figs. 6, 7, and 8. Fig. 6 depicts the total energy consumption when the number of sensor nodes varies. The energy consumption increases with the number of nodes since more nodes are involved in packet forwarding. On the other hand, this figure shows that the energy consumption for the proposed algorithm is less than that in VBF and HH-VBF routing protocol only on dense networks, when the number of nodes is greater than 300 nodes, indicating that the CVBF and the BS-CVBF algorithm can save more energy with high node density.

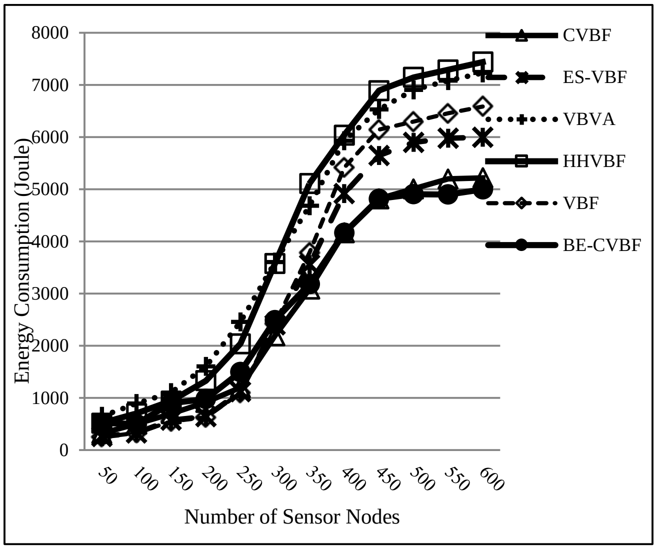


Figure 6. BS-CVBF energy consumption *vs.* number of sensor nodes.

Figure 7 shows the packet delivery ratio with the number of sensor nodes. It is seen that the packet delivery ratio increases with the increase of the number of nodes. When more than 200 nodes are deployed in the space, the packet delivery ratio remains above 90% for both ES-VBF routing protocol and BS-CVBF algorithm. Figure 7 shows that our algorithm gives better results in packet delivery ratio than VBF, HHVBF, VBVA, ES-VBF, and CVBF protocols.

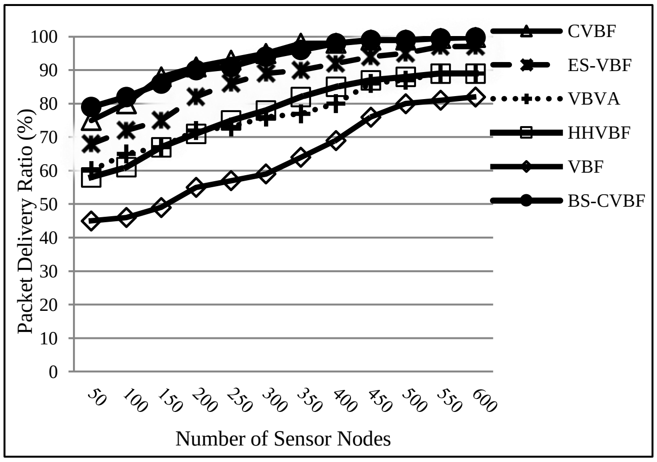


Figure 7. BS-CVBF packet delivery ratio *vs.* number of sensor nodes.

Figure 8 describes the average end-to-end delay with the number of sensor nodes. It is seen that the average end-to-end delay decreases with the increase of node density in the network. When the number of sensor nodes increases, the paths from the source to the sink are closer to the optimal path (α=0); therefore, the average end-to-end delay decreases. While Fig. 9 shows the CPU utilization with respect to the number of node.

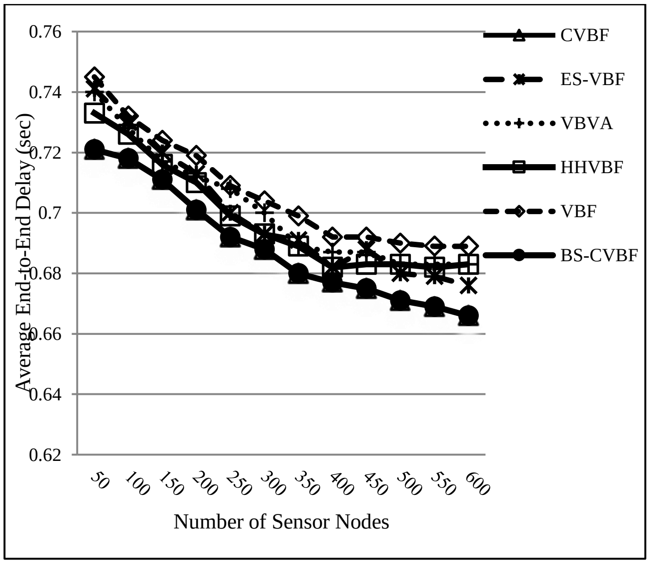


Figure 8. BS-CVBF average end-to-end delay *vs.* number of sensor nodes.

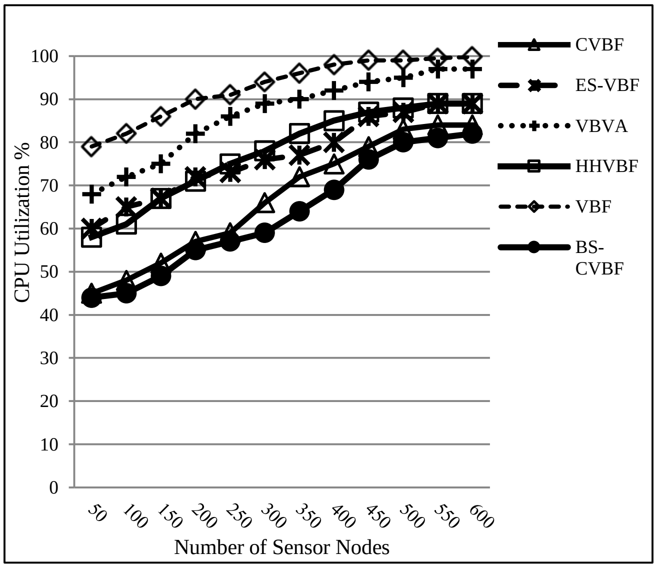


Figure 9. BS-CVBF CPU utilization *vs.* number of sensor nodes.

We evaluate the performance of BS-CVBF under various network scenarios. The simulation results show that CVBF significantly exhibits a better performance than VBF, HH-VBF, VBVA, and ES-VBF protocols since it has: lower energy consumption, higher packet delivery ratio, and lower average end-to-end delay.

Calculating the cluster width *cw* depends on two parameters: the surface area of the network X×Y and choosing the number of clusters *N*. We choose a value of *N* for which the cluster width is nearest to the value of $\sqrt{2}$R. We conclude this after examining different values of *N*. This is because each node can transmit the data packets only to the neighbors allocated in its transmission range.

V. PERFORMANCE UNDER VARYING NUMBER OF CLUSTERS

In this section, we vary the number of clusters, N, from 2^0 to 6^2 (N=1, 4, 9, 16, 25, and 36), keeping the network surface area is 600×600 m2 and R is 100m. We need to prove that the optimal value of N is the value which makes cw as near as possible to $\sqrt{2}$R. Table II shows the different values of cluster width based on equation (4) and the number of clusters.

TABLE II. CLUSTER WITH VALUES UNDER VARYING NUMBER OF CLUSTERS

Number of Clusters (*N*)	**Cluster Width (*cw*)**	**$\sqrt{2}$R (R=100)**
1	600	141.42
4	300	
9	200	
16	**150**	
25	120	
36	100	

Table II shows that the nearest value of the cluster width in this experiment is 150; this means that the preferred number of the clusters is 16 clusters. Figures 10, 11, and 12 show our simulation results in the different values of *N*, 1, 4, 9, 16, 25 and 36. Our simulation results give better performance when *N* is 16 because *cw*=150 m is the nearest value to $\sqrt{2}$R (141.42 m). It gives lower energy consumption, higher packet delivery ratio, and lower average end-to-end delay.

Figure 10 depicts the total energy consumption of the proposed BS-CVBF algorithm as the number of clusters varies. The energy consumption increases with the number of nodes since more nodes are involved in packet forwarding. On the other hand, this figure shows that the energy consumption for the proposed algorithm when the number of clusters is 16 (*N*=16) is less than the other values of *N*. This means that dividing the network into 16 clusters gives lower energy consumption.

Figure 11 shows the packet delivery ratio with the number of sensor nodes under varying the clusters number. It is seen that the packet delivery ratio increases with the increase of the number of clusters till 16 clusters and the packet delivery ratio decreases at 25 clusters.

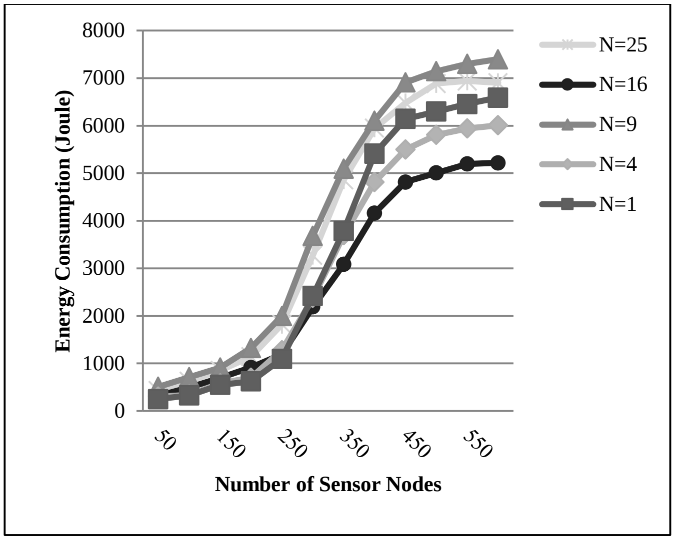


Figure 10. BS-CVBF energy consumption under varying clusters number.

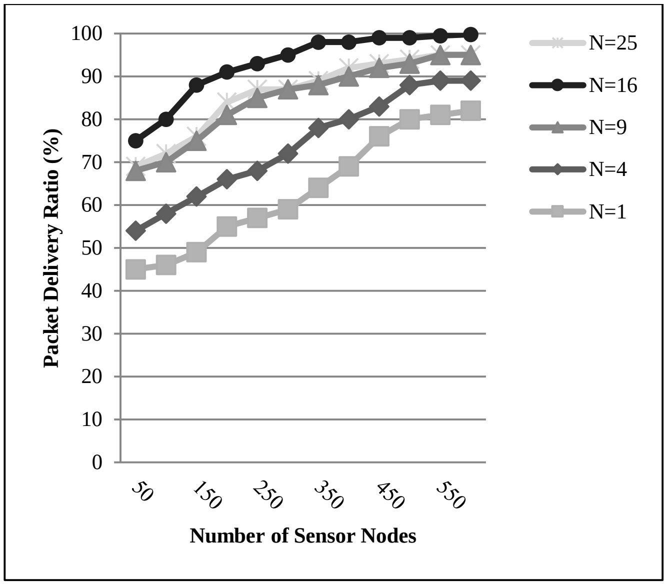


Figure 11. BS-CVBF packet delivery ratio under varying clusters number.

Figure 12 describes the average end-to-end delay with the number of sensor nodes. It is seen that the average end-to-end delay decreases with the increase of number of clusters in the network. When the number of cluster exceeds 16 clusters the average end-to-end delay increases again. Therefore, choosing *N*=16 gives better performance than that for *N*=1, 4, 9, 25.

The simulation results in Figs. 10, 11, and 12 give better performance when *N* is 16 because *cw*=150 m is the nearest value to $\sqrt{2}$R (141.42 m). It gives lower energy consumption, higher packet delivery ratio, and lower average end-to-end delay for the proposed algorithm.

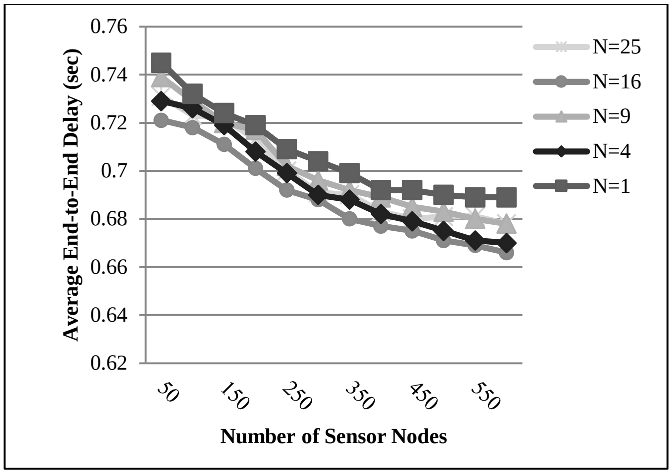


Figure 12. BS-CVBF average end-to-end delay under varying clusters number.

VI. CONCLUSIONS

In this paper, we propose a bounded side-based clustering vector-based forwarding algorithm to improve the performance of the location-based routing protocol in underwater wireless sensor networks. In the proposed algorithm, the whole network is divided into a predefined number of clusters and determining an internal bounded side with its own coordinates for each cluster. The main target of our proposed algorithm is to avoid the calculations taken for the nodes in which they never move away from its cluster. These nodes' positions are near the cluster axis and far enough from the cluster boundaries. For this reason, we separate between the nodes that can move outside the clusters and the nodes that never move away. Simulation results demonstrate that the proposed algorithm reduces the energy consumption especially in dense networks, increases the packet delivery ratio especially in sparse networks, and decreases the average end-to-end delay in both sparse and dense networks. These advantages are emphasized when the algorithm is compared with five other powerful routing algorithms: VBF, Hop-by-Hop VBF (HH-VBF), Vector-Based Void Avoidance (VBVA), Energy-Saving VBF (ES-VBF), and the CVBF routing protocols. It is interesting to note that our multiple-cluster algorithm is a good generalization to the VBF protocol. The VBF results from our algorithm by adopting the special case of single-cluster manipulation.

REFERENCES

[1] Dina M. Ibrahim, Mohmoud M. Fahmy, Tarek E. ElTobely, and ElSayed A. Sallam, “Enhancing the Vector-Based Forwarding Routing Protocol for Underwater Wireless Sensor Networks: A Clustering Approach,” The Tenth International Conference on Wireless and Mobile Communications (ICWMC 2014), Seville, Spain, June, 2014, pp. 98-104.

[2] E. M. Sozer, M. Stojanovic, and J. G. Proakis, “Underwater Acoustic Networks,” IEEE Journal of Ocean Engineering, vol. 25, Jan. 2000, pp. 72-83.

[3] S. Climent, A. Sanchez, J. V. Capella, N. Meratnia, J. J. Serrano, “Underwater Acoustic Wireless Sensor Networks: Advances and Future Trends in Physical, MAC and Routing Layers” International Journal of Sensors (Basel), vol. 14, no. 1, 2014, pp. 795–833.

[4] M. Garcia, S. Sendra, M. Atenas, and J. Lloret, “Underwater Wireless Ad-hoc Networks: A Survey,” International Journal of Mobile Ad hoc Networks: Current Status and Future Trends, 2011, pp. 379-411.

[5] K. Shazzad, T. Kemal, and A. Esam, "Multi-Hop-Enabled Energy-Efficient MAC Protocol for Underwater Acoustic Sensor Networks," International Journal of Sensor and Actuator Networks, vol. 4, no. 3, 2015, pp. 226-250.

[6] M. Ayaz, I. Baig, A. Abdullah, and I. Faye, “A Survey on Routing Techniques in Underwater Wireless Sensor Networks,” Journal of Network and Computer Applications, vol. 3, no. 4, 2011, pp. 1908-1927.

[7] J. Lloret, M. Garcia, S. Sendra, and G. Lloret, ”An underwater wireless group-based sensor network for marine fish farms sustainability monitoring,” International Journal of Telecommunication Systems, 2014, pp. 1-18.

[8] M. Ayaz and A. Abdullah, “Underwater Wireless Sensor Networks: Routing Issues and Future Challenges,” Proc. of the 7th International Conference on Advances in Mobile Computing and Multimedia. ACM, Malaysia, 2009, pp. 370-375.

[9] I. F. Akyildiz, D. Pompili, and T. Melodia, “Underwater Acoustic Sensor Networks: Research Challenges,” Ad Hoc Networks (Elsevier), vol. 3, no. 3, Mar. 2005, pp. 257–279.

[10] D. Pompili, T. Melodia, and I. F. Akyildiz, “Distributed Routing Algorithms for Underwater Acoustic Sensor Networks,” IEEE Trans. Wireless Communications, vol. 9, no. 9, Sept. 2010, pp. 2934-3944.

[11] P. Xie, J.-H. Cui, and L. Lao, “VBF: Vector-based Forwarding Protocol for Underwater Sensor Networks,” International Conference on Networking (IFIP networking), 2006, pp. 1-20.

[12] N. Nicolaou, A. See, P. Xie, J.-H. Cui, and D. Maggiorini, “Improving the Robustness of Location-based Routing for Underwater Sensor Networks,” Proc. Of the OCEANS’07, Europe, June 2007, pp. 1-6.

[13] A. Sharma and H. A. Gaafar, “A Survey on Routing Protocols for Underwater Sensor Networks,” International Journal of Computer Science & Communication Networks,vol. 2, no. 1, 2012, pp. 74-82.

[14] P. Xie, Z. Zhou, Z. Peng, J.-H. Cui, and Z. Shi, “Void Avoidance in Three-dimentional Mobile Underwater Sensor Networks,” Proc. of the 4th International Conference of Wireless Algorithms, System, and Applications (WASA 2009), USA, August 2009, pp. 305-314.

[15] W. Bo, L. Yong-mei, and J. Zhigang, “ES-VBF: An Energy Saving Routing Protocol,” Proc. of the 2012 International Conference on Information Technology and Software Engineering, 2012, pp. 87-97.

[16] E. A. Carlson, P.P. Beaujean and E. An., “Location-aware Routing Protocol for Underwater Acoustic Networks,” Proc. of the OCEANS, 2006, pp. 1-6.

[17] R. Thumpi, R.B. Manjula, and S.M. Sunilkumar, “A Survey on Routing Protocols for Underwater Acoustic Sensor Networks,” International Journal of Recent Technology and Engineering (IJRTE), vol. 2, May 2013, pp. 170-175.

[18] NS-2: Network Simulator, http://www.isi.edu/nsnam/ns, last access: 30th August 2015.

[19] P. Xie, et al., “Aqua-sim: A Ns-2 based Simulator for Underwater Sensor Networks,” Proc. of IEEE/MTS OCEANS, 2009, pp. 1-7.

[20] LinkQuest, http://www.link-quest.com, last access: 30th August 2015.