Improving the Robustness of Location-Based Routing for Underwater Sensor Networks

Nicolas Nicolaou†, Andrew See†, Peng Xie†, Jun-Hong Cui†, Dario Maggiorini‡
(nicolas, ags03001, xp, jcui)@engr.uconn.edu, dario@dico.unimi.it
† Computer Science & Engineering Department, University of Connecticut, Storrs, CT 06029
‡ Computer Science Department, University of Milano, via Comelico 39, 20135 Milano, Italy

***Abstract*— This paper investigates a fundamental networking problem in underwater sensor networks: robust and energy-efficient routing. We present an adaptive location-based routing protocol, called *hop-by-hop vector-based forwarding* (HH-VBF). It uses the notion of a "routing vector" (a vector from the source to the sink) acting as the axis of the "routing pipe", similar to the vector based forward (VBF) routing in [11]. Unlike the original VBF approach, however, HH-VBF suggests the use of a routing vector for each individual forwarder in the network, instead of a single network-wide source-to-sink routing vector. By the creation of the hop-by-hop vectors, HH-VBF can overcome two major problems in VBF: (1) too small data delivery ratio for sparse networks; (2) too sensitive to "routing pipe" radius threshold. We conduct simulations to evaluate HH-VBF, and the results show that HH-VBF yields much better performance than VBF in sparse networks. In addition, HH-VBF is less sensitive to the routing pipe radius threshold. Furthermore, we also analyze the behavior of HH-VBF and show that assuming proper redundancy and feedback techniques, HH-VBF can facilitate the avoidance of any "void" areas in the network.**

***Index Terms*— Underwater Sensor Networks; Location-Based Routing; Energy Efficiency; Robustness; Avoid "Voids"**

I. INTRODUCTION

Recently, there has been growing interest in applying sensor networks into underwater environments (i.e., building underwater sensor networks) to enable/enhance applications such as oceanographic data collection, pollution monitoring, offshore exploration and tactical surveillance applications [9], [8], [2], [4], [3], [7]. In underwater environments, due to water absorption, radio does not work well. Thus acoustic communication is usually employed as a viable solution in underwater sensor networks. However, due to the physical characteristics of sound signals, acoustic channels are featured with low available bandwidth, very large propagation delay, and very high error probability. Another uniqueness in underwater environments is that most sensor nodes could be passively mobile with water currents (this setting is desirable for many applications, such as estuary water monitoring and submarine detection [3]).

To design an autonomous underwater sensor network, all the aforementioned new features should be taken into consideration. One of the key functionalities in building underwater sensor networks is to route data from sources (sensor nodes which collect and generate data) to sinks (some surface nodes which are connected to on-shore command centers). The first routing protocol designed for mobile underwater sensor networks is Vector Based Forwarding (VBF), which was proposed in [11]. VBF is a trajectory-based forwarding protocol. It represents a trajectory with a "routing vector" from the source to the sink. Intuitively a *virtual pipe* with the source-to-sink vector as the axis is used as the abstract route for data delivery. If the pipe is "populated" by nodes then the data packets can be forwarded to the sink. The radius of the virtual pipe is a predefined distance threshold. For any sensor node which receives data, it first computes its distance to the routing vector. If this distance is smaller than the threshold, then the node is considered as a *candidate* to forward the data. Otherwise, the node simply discards the data. To reduce the traffic in dense networks, VBF adopts a distributed self-adaptation algorithm, in which all the candidate nodes are coordinated and finally only several most "desirable" ones can forward the data packets. Compared with naive flooding, VBF can significantly reduce network traffic, thus saving energy. It is also robust to topology dynamics since it is a location-based on-demand routing protocol, and no pre-computed routes maintained in sensor nodes.

However, there are two major drawbacks with VBF: (1) Because of the use of the unique source-to-sink vector, the creation of a single virtual pipe may significantly affect the routing efficiency in different node density areas. If nodes in one area are too sparsely distributed, then it is quite possible that very few or even no nodes lie within the virtual pipe eligible for data forwarding, as may lead to network disconnection, hence data delivery ratio is degraded; (2) Again because of the single source-to-sink vector design, VBF is too sensitive to the routing pipe radius threshold. As shown in [11], the routing pipe radius threshold significantly affects the routing performance, as may not be a desirable feature in the real protocol deployment.

To overcome these problems in VBF, in this paper, we present a protocol, called **Hop-by-Hop Vector-Based Forwarding (HH-VBF)**. It uses the same concept of routing vector as VBF. However, instead of using a single virtual pipe from the source to the sink, HH-VBF defines a different virtual pipe around the per-hop vector from each forwarder to the sink. In this way, each node can adaptively make packet forwarding decisions based on its current location. This design can directly bring the following benefits: (1) Since each node has its own routing pipe, the maximum pipe radius is the transmission range. In other words, there is no necessity to increase the pipe radius beyond the transmission range in order to enhance routing performance; (2) In sparse networks, though the number of eligible nodes may be small, HH-VBF can find a data delivery path as long as there exists one in the network. Thus, HH-VBF enhances data delivery ratio in sparse networks compared with VBF. e conduct simulations to evaluate HH-VBF, and the results show that HH-VBF yields much better performance than VBF in sparse networks. In addition, HH-VBF is less sensitive to the routing pipe radius threshold. Furthermore, we also analyze the behavior of HH-

VBF and show that assuming proper redundancy and feedback techniques, HH-VBF can facilitate the avoidance of any "void" areas in the network.

The rest of this paper is organized as follows. In Section II, we describe some background information and give a brief review on VBF. Then, in Section III, we present our new protocol HH-VBF, and analyze its benefits over VBF. After that, we report our simulation results in Section IV, and conclude our paper in Section V.

II. BACKGROUND

In this section, we briefly describe some unique features of underwater sensor networks which are closely related to the routing protocol design. We also give a brief review on VBF to lay the foundation for our proposal.

A. Uniqueness of Underwater Sensor Networks

Compared with terrestrial sensor networks, underwater sensor networks have many unique features, which pose many new challenges for the design and implementation of network protocols. In the following, we list some key features which significantly affect routing protocols.

1) Sensor Nodes: Similar to terrestrial sensor nodes, underwater sensor nodes are generally powered by batteries [1], [4], [3]. This makes it important for protocols to be energy efficient. Thus, routing algorithms should reduce the traffic in the network as much as possible, and, of course, maintain a high successful data delivery ratio at the same time.

In addition, underwater sensor nodes should be equipped with acoustic modems since radio does work well in water due to quick absorption and heavy attenuation. Acoustic communication poses many challenges. First, the available bandwidth of underwater acoustic channels is limited and dramatically depends on both transmission range and frequency. According to [5], nearly no research and commercial system can exceed 40 `km` $\times$ `kbps` as the maximum attainable `Range` $\times$ `Rate` product. Second, the propagation speed of sound is much smaller than that of radio. To be specific, the speed of sound in water can reach at most 1.5×10^3 m/s, while radio can propagate with a speed of 3×10^8 m/s. This results in very large propagation delay. Moreover, underwater acoustic channels are affected by many factors, such as path loss, noise, multi-path, and Doppler spread. All these cause high error probability in acoustic channels. Therefore, an effective routing protocol should also take low bandwidth, large propagation delay, and high error probability into account.

2) Network Topology: In contrast to terrestrial sensor networks which are usually studied in a 2D environment, underwater sensor networks work naturally in a 3D world. This is due to the fact that underwater nodes can float with water current on top of other deployed nodes. In a mobile 3D underwater sensor network scenario, it is usually assumed that sensor nodes are deployed in layers at certain depths (which can be realized by installing a buoyancy device to each sensor node) [3]. At each layer, nodes can float with water current, with a speed of 1-3 m/s. Vertically, a node may have a small depth variation, which is usually negligible.

Due to the higher dimensional network topology, a good routing protocol should be more careful to control the number of nodes involved, since even a single forwarding may lead a lot of nodes to overhear the data. Moreover, node mobility should be taken into consideration.

B. Overview of VBF

Xie etc. proposed a first routing protocol, VBF, for underwater sensor networks [10]. VBF is similar to trajectory-based forwarding with the trajectory function limited to a source-to-sink vector. In VBF, each node in the network is assumed to know its location, and each packet carries the locations of the source, the sink, and the sender. The main idea of the protocol is to use a virtual routing pipe, with the source-to-sink vector as its axis and Th as its radius (where Th is a predefined threshold). If a node lies within this routing pipe, it forwards the packets from the source. More specifically, each intermediate node N_i that receives the source data computes its distance d_i from the source-to-sink vector. If $d_i \leq Th$, then node N_i forwards the packet, otherwise, N_i discards the packet. The major advantage of VBF is that there is no need for route planning or propagation, which is robust to the network dynamics caused by node mobility. Fig. 1 illustrates the basic idea of VBF. In the figure, node S_1 is the source, and node S_0 is the sink. The routing vector is specified by $\overrightarrow{S_1S_0}$. 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 (i.e., the distance threshold).

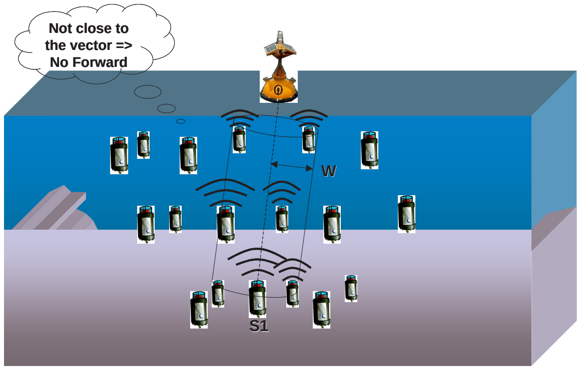

Fig. 1. A high-level view of VBF for underwater sensor networks.

As we can see, VBF tends to minimize the traffic in the network by specifying the closest path to the sink. However, when sensor nodes are densely deployed, VBF may involve too many nodes in data forwarding, which in turn increases the energy consumption. Thus, it is desirable to adjust the forwarding policy based on the node density. VBF adopts a self-adaptation algorithm to allow each node to estimate the density in its neighborhood (based on local information) and forward packets adaptively.

1) The Self-Adaptation Algorithm: In VBF, an important notation **desirableness factor** is introduced to measure the "suitableness" of a node to forward packets,

Definition 1: Given a routing vector $\overrightarrow{S_1S_0}$, where S_1 is the source and S_0 is the sink, for forwarder F, the **desirableness factor**, α, of a node A, is defined as

$$\alpha = \frac{p}{W} + \frac{(R - d \times cos\theta)}{R},$$

where p is the distance from A to the routing vector $\overrightarrow{S_1S_0}$, d is the distance between node A and node F, and θ is the angle between $\overrightarrow{FS_0}$ and $\overrightarrow{FA}$. R is the transmission range and W is the radius of the "routing pipe".

Fig. 2 depicts the various parameters used in the definition of desirableness factor. From the definition, we can easily get

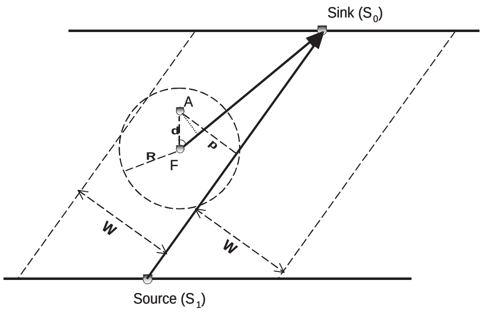

Fig. 2. An illustration of desirableness factor in VBF.

that for any node close enough to the routing vector, i.e., $0 \leq p \leq W$, the desirableness factor of this node is in the range of $[0, 3]$. For a node, the smaller the desirableness factor, the higher the priority to forward the packets.

The self-adaptation algorithm works as follows: when a node receives a packet, it first determines if it is eligible for packet forwarding (i.e., close enough to the routing vector). If yes, it holds the packet for a time period, $T_{adaptation}$, which is computed based on its desirableness factor and other network parameters. During the packet holding time period $T_{adaptation}$, if the node receives duplicate packets from n other nodes, then it has to compute its desirableness factors relative to these nodes, $\alpha_1, \ldots, \alpha_n$, and the original forwarder, α_0. It then weighs these desirableness factors. If the minimum one is even smaller than a predefined threshold, it will forward the packet; otherwise, it will discard the packet. Essentially, this self-adaptation algorithm gives higher priority to the desirable node to continue forwarding the packet, and it also allows a less desirable node to have chances to re-evaluate its "importance" in its neighborhood. If there are many other more desirable nodes for the packet, there is no necessity for it to forward it any more. In real implementation, if a node receives more than two duplicate packets during its waiting time, it is most likely that this node will not forward the packet.

C. Drawbacks of VBF

By introducing the self-adaptation algorithm, VBF can significantly reduce the traffic in dense networks. However, there are two major drawbacks with VBF:

(1) VBF limits the routing-involved nodes within a single source-to-sink routing pipe by a predefined radius. In sparse networks, if no nodes lie within this pipe, then data packets can not be forwarded to the sink even though paths may exist outside the pipe. In VBF, these paths will not be discovered and thus delivery ratio will be severely affected. Fig. 3 shows possible effects of VBF with one fixed routing pipe for each source. In this example, packets from nodes A and C are unable to reach the sink because no nodes lie within the pipe, though a path does exist through other nodes.

(2) Because of the use of a single source-to-sink routing vector, VBF is very sensitive to the routing pipe radius threshold. As shown in [10], in general, the bigger the radius is, the higher successful data delivery ratio VBF can achieve, and the more optimal path can VBF select. Thus, in a network with uneven node distribution, it is difficult to choose a proper routing pipe radius threshold. However, in underwater environments, uneven node distribution is quite common due to node mobility and environmental activities (such as shipping and fishery). Therefore, some measures should be taken to make VBF a practical solution.

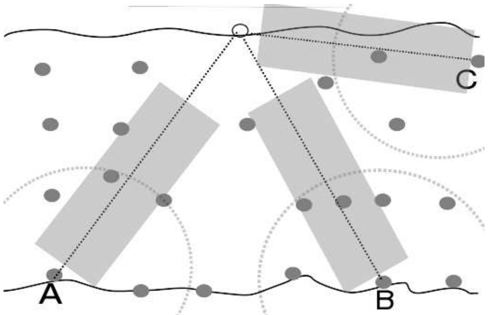

Fig. 3. VBF using single routing pipe for each source. The gray rectangles show the areas within the routing pipes. The transmission ranges of the three sources are shown by dotted circles.

III. HOP-BY-HOP VECTOR-BASED FORWARDING (HH-VBF)

In this section, we present our new protocol HH-VBF, examine how it overcomes the problems of VBF, and pinpoint other potential benefits it can bring.

A. HH-VBF Protocol Overview

In HH-VBF, we redefine the routing virtual pipe to be a per-hop virtual pipe creation, instead of a unique pipe from the source to the sink. This hop-by-hop approach allows the expansion of the probability of finding a routing path in comparison with VBF. Consider a node N_i which receives a packet from the source or a forwarder node S_j. Upon receipt of the packet, the node computes the vector from the sender S_j to the sink. In this way, the forwarding pipe changes each hop in the network, giving the name **hop-by-hop vector based forwarding (HH-VBF)**. After a receiver computes the vector from its sender to the sink, it calculates its distance to that vector. If this distance is smaller than the predefined threshold then it is eligible to forward the packet, and we refer to such a node as a *candidate forwarder* for the packet.

As in VBF, each candidate forwarder maintains a self-adaptation timer which depends on the desirableness factor. The timer represents the time the node holds the packet before forwarding it. We modify Definition 1 and get a new definition of the desirableness factor for HH-VBF:

Definition 2: For a candidate forwarder F, the **desirableness factor**, $\alpha\prime$, of a node A, is defined as

$$\alpha\prime = \frac{(R - d \times cos\theta)}{R},$$

where d is the distance between node A and node F, and θ is the angle between $\overrightarrow{FS_0}$ and $\overrightarrow{FA}$. R is the transmission range and S_0 is the sink.

The self-adaption algorithm in HH-VBF is different from that in the original VBF. As we recall, due to the effective packet suppression strategy adopted in VBF, only a few paths could be selected to forward packets. This may cause problems in sparse networks. To enhance the packet delivery ratio in sparse networks, we introduce some redundancy control in the self-adaption procedure for HH-VBF.

In HH-VBF, when a node receives a packet, it first holds the packet for some time period proportional to its desirableness factor (this is similar to VBF). Therefore, the node with the smallest desirableness factor will send the packet first. Following this way, each node in the neighborhood may hear the same packet multiple times. HH-VBF allows each node overhearing the duplicate packet transmissions to control the forwarding of this packet as follows: the node calculates its distances to the various vectors from the packet forwards to the sink. If the minimum one of these distances is still larger than a pre-defined minimum distance threshold β, this node will forward the packet; otherwise, it simply drops the packet. Obviously, the bigger β is, the more nodes will be allowed for packet forwarding. Thus, we HH-VBF can control forwarding redundancy by adjusting β.

Each node that qualifies as a candidate forwarder delays the packet forwarding by an interval $T_{adaptation}$ which is computed the same way as in VBF. Then each node still uses the self-adaptation algorithm to limit the redundant packets.

Fig. 4 illustrates a high level picture of HH-VBF using the same network setting as in Fig. 3. As we can see, in HH-VBF, nodes A and C can reach the sink by using paths that are not possible with VBF.

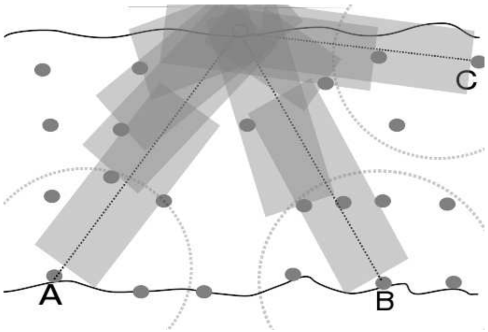

Fig. 4. HH-VBF with per-hop vector computing for the same network networking as in Fig. 3.

B. Analysis of HH-VBF

In this section we examine the major benefits of HH-VBF over its predecessor, VBF. We also discuss how HH-VBF helps to avoid "void", i.e., routing holes, in networks.

1) HH-VBF vs VBF: Compared with VBF, the major innovation of HH-VBF is the hop-by-hop approach. Though the basic idea is simple, it can bring two significant benefits: (1) HH-VBF can find more paths for data delivery in sparse networks; (2) HH-VBF is less sensitive to the routing pipe radius (i.e., the distance threshold). Correspondingly, we have the following two lemmas.

Lemma 1: Given the same routing pipe radius, if a packet is routable in VBF, then it must be routable in HH-VBF.

Proof: If we can show that any routing-involved node in VBF is also involved in routing in HH-VBF, then we prove the lemma. Now, we assume that in HH-VBF a node N_i is not involved in routing. This implies that in the network there is no path leading from the source to N_i give the distance threshold. Thus, the source-to-sink routing pipe does not cover node N_i, that is, N_i is not involved in routing. Using the contradiction method, we prove the lemma. ■

Lemma 1 indicates that HH-VBF is at least as reliable as VBF.

Lemma 2: The valid range of routing pipe radius of HH-VBF is $[0, R]$, while the valid range of VBF is $[0, D]$, where R is the node transmission range, and D is the network diameter (here we assume all nodes have the same transmission range).

Proof: In HH-VBF, each node makes packet forwarding decisions based on its distance to the vector from its forwarder to the sink. If the distance is bigger than the predefined pipe radius, the node will forward the packet, otherwise it will discard the packet. In this way, when the pipe radius is bigger than the transmission range of the forwarder, those nodes which are outside the transmission range while still lie in the routing pipe are useless since they can not hear the packets from the forwarder. Thus, the valid range of routing pipe radius of HH-VBF is $[0, R]$, where R is the transmission range.

In VBF, each node makes packet forwarding decisions based on its distance to the vector from the source to the sink. When the pipe radius is bigger than the transmission range, those nodes which are outside the transmission range of one forwarder while still lie in the routing pipe may hear packets from other forwarder. This means that they may be still eligible for packet forwarding. Thus, theoretically there is no upper limit for the pipe radius of VBF, while in practice, the valid range of routing pipe radius of VBF is $[0, D]$, where D is the network diameter. ■

From [10], we know that, the bigger the pipe radius, the higher successful data delivery ratio VBF can achieve, and the more optimal the paths VBF can select. Thus, for networks with different density, a proper pipe radius should be carefully chosen. While for HH-VBF, from Lemma 2, we can see that the biggest value of the pipe radius is R, which will clearly yield the highest successful data delivery ratio. Thus, in HH-VBF, we can eliminate the trouble of tuning the pipe radius by simply choosing the transmission range R.

2) Avoid "Void": We consider scenarios where some areas of the network are not populated with nodes, i.e. there exit "voids" in the network. In the following, we show how HH-VBF could facilitate the avoidance of such situations.

One may notice that when the the pipe radius is set to the node transmission range, HH-VBF without self-adaptation algorithm in fact becomes pure flooding. Since flooding is guaranteed to find a path to the sink if it exists, HH-VBF essentially includes all possible paths in the network. This means that HH-VBF with self-adaptation is guaranteed to route packets successfully if the number of allowed redundant forwarders is sufficiently high. However, due to the choice of many parameters in the self-adaptation algorithm, HH-VBF may not be able to route if there are voids in the network. In such case, a forwarder is unable to reach any node other than the previous hop. We argue that with a feedback mechanism, HH-VBF could be easily modified to detect and avoid such voids in the network.

IV. SIMULATION RESULTS

In this section we conduct simulations to evaluate the performance of HH-VBF, compared with that of VBF.

A. Simulation Setting

We use NS-2 to simulate 3D underwater sensor networks. A routing layer agent is added to simulate HH-VBF and VBF. An application layer agent is used to simulate the traffic source and sink. We use the same broadcast MAC protocol as in [10]. In this MAC protocol, when a node has packets to send, it first senses the channel. If the channel is free, it broadcasts the packets. Otherwise, it backs off. The packet will be dropped if the node backs off 4 times. Since there is no collision resolution in this broadcast MAC protocol, we mitigate the effect of packet collisions by adopting the self-adaption algorithms and using a low data generation rate. In our simulations, we set the data generation rate as 1 packet every 10 seconds, which can help to effectively avoid the interference between two continuous data packets. As to acoustic communications, we set the parameters similar to a commercial acoustic modem, LinkQuest UWM1000 [6]: the bit rate is $10k$ bps; the transmission range is 100 meters; and the energy consumptions in sending mode, receiving mode and idle mode are $2w$, $0.75w$ and $8mw$ respectively. Further, we set the packet size to 50 Bytes, the pipe radius to 100 meters (the same as the transmission range), and β is set to 75 m for HH-VBF.

In all the simulation experiments described in this section, sensor nodes are randomly distributed in a 3D field of $1000m \times 1000m \times 500m$. There are one data source and one sink. The source is fixed at location $(900, 900, 500)$ near one corner of the field at the floor, while the sink is at location $(100, 100, 0)$ near the opposite corner at the surface. Besides the source and the sink, all other nodes are mobile as follows: they can move in horizontal two-dimensional space, i.e., in the X-Y plane (which is the most common mobility pattern in underwater applications). Each node randomly selects a destination and moves toward that destination. Once the node arrives at the destination, it randomly selects a new destination and moves in a new direction. For each test, the results are averaged over 50 runs, with a randomly generated topology in each run. The total simulation time for each run is 1000 seconds.

Performance Metrics We propose three metrics: *success rate*, *energy cost* and *energy tax*. *Success rate* is defined as the ratio of the number of packets successfully received by the sink to the number of packets generated by the source. *Energy cost* is measured by the total energy consumption of all the nodes in the network. *Energy tax* is defined as the average energy consumption for each successfully received packet.

B. Results and Analysis

1) The Impact of Node Density: In this set of simulations, we examine the impact of node density. We fix the node speed at 0 (i.e., static networks), and change node density by varying the number of nodes deployed in the field from 500 to 3000. The results for success rate, energy cost and energy tax are plotted in Fig. 5, Fig. 6, and Fig. 7 respectively.

From Fig. 5, we can clearly observe the general trend of success rate for both VBF and HHVBF: with the increasing node density, the success rate is enhanced. This is intuitive: for any node in the network, as the network density becomes larger, more nodes will fall in its routing pipe (with fixed radius as the transmission range). In other words, more nodes are qualified for packet forwarding, as naturally leads to higher success rate. Future, we can see that the success rate of HH-VBF is significantly improved upon VBF, especially when the network is sparse. This observation is consistent with our early analysis: HH-VBF can find more paths for data delivery in sparse networks.

Fig. 6 shows us that the energy cost of HH-VBF is higher than that of VBF, and the gap becomes more significant as the network gets denser. This is reasonable as the higher the node density, the more paths HH-VBF can find. We normalize the energy consumption, i.e., compute the energy tax, and the results are illustrated in Fig. 7. From this figure, we can observe that when the network is sparse, the normalized energy cost of HH-VBF is greatly lower than that of VBF. For example, when the number of nodes is 1000, the energy tax of HH-VBF is 226 J/pkt, while the energy overhead of VBF is as high as 4919 J/pkt. This is mainly because the data delivery ratio of VBF is extremely low (2% when the network size is 1000). This further confirms that VBF is not good for sparse networks. On the other hand, when the network gets denser, VBF shows its advantage over HH-VBF: HH-VBF still tends to find more paths, while the delivery ratio has reached the maximum. In this case, more paths do not help to increase the success rate, but more energy cost will be introduced. Thus, we believe it is worth investigating an adaptive scheme for unevenly distributed networks, exploring both the benefits of VBF and HH-VBF. We leave this study as our future work.

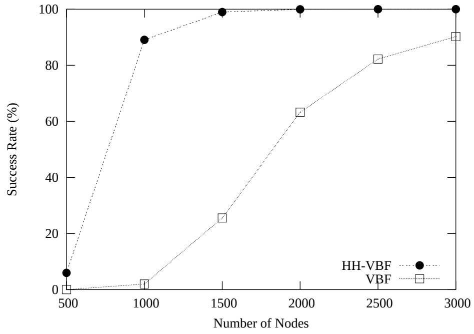

Fig. 5. Success rate vs node density.

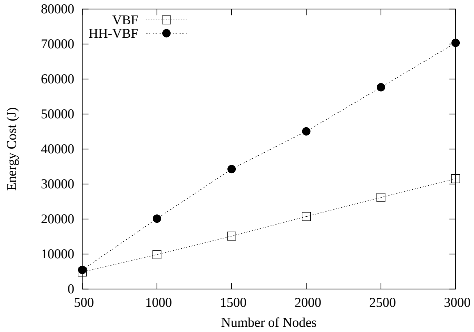

Fig. 6. Energy cost vs node density.

2) The Impact of Node Mobility: In this set of simulations, we explore how node mobility impacts the performance of HH-VBF. We fix the network size at 1000 (a relatively sparse network), and vary the node speed from 0 to 3 m/s. Fig. 8, Fig. 9, and Fig. 10 plot the results for the three metrics.

From Fig. 8, we can observe that the node mobility has different effects on the success rate of VBF and HH-VBF when the node speed is low. By conducting many additional simulation experiments, we find this is mainly due to the randomness of network topology generation. For VBF, when node pattern changes from "static" to "mobile", the mobility

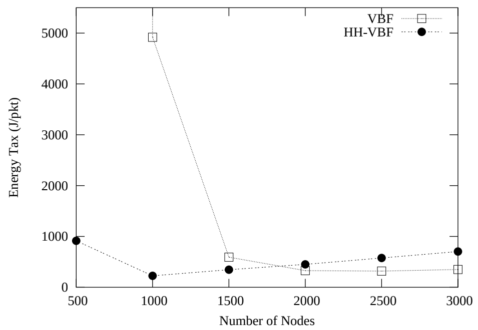

Fig. 7. Energy tax vs node density.

actually helps to increase the chance that non-connected paths become connected, while for HH-VBF, since there are more routing pipes in the network, light node mobility causes the chance that non-connected paths become connected smaller. In fact, when the network is extremely sparse, e.g., the network size is 500 in our simulations, the impact of light node mobility on HH-VBF has the same trend for VBF: the success rate is slightly enhanced. In addition, when we increase the number of simulation runs, the effect of node mobility is decreased (due to space limit, these results are not shown in in this paper). Furthermore, from Fig. 8, we can see that as the node speed gets higher, the success rate of both VBF and HH-VBF becomes stable. This indirectly confirms that experiencing more topologies will help eliminate the difference caused by the topology randomness.

Fig. 9, Fig. 10, and Fig. 8 together convey the major information: both HH-VBF and VBF are robust to node mobility, while HH-VBF has much better performance (in terms of both success rate and energy tax) than VBF in sparse networks.

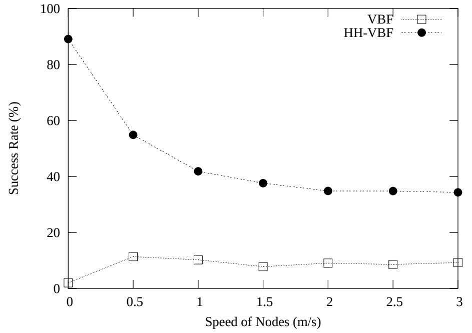

Fig. 8. Success rate vs node speed.

V. CONCLUSIONS AND FUTURE WORK

This paper has presented, HH-VBF, an enhanced version of the VBF routing protocol for Underwater Sensor Networks. The new proposal introduces a hop-by-hop approach, which is simple while novel, and it can significantly improve the robustness of packet delivery in sparse networks: enhancing the data delivery ratio while taxing less energy.

Future Work: We would conduct future studies in the following two directions: 1) We plan to explore an adaptive design for unevenly distributed networks, making best use of the advantages of VBF and HH-VBF; 2) We plan to add a feedback mechanism to the HH-VBF algorithm to detect and avoid voids in the network. By increasing the delay for the nodes on the path toward the void, the algorithm can choose alternate paths automatically.

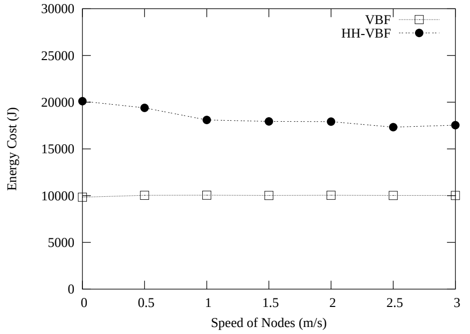

Fig. 9. Energy cost vs node speed.

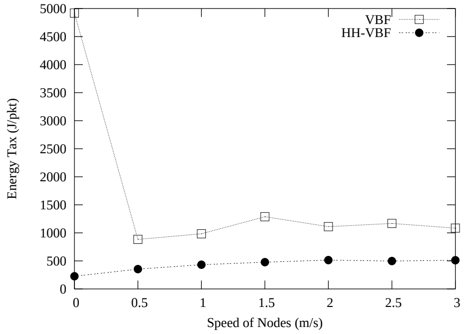

Fig. 10. Energy tax vs node speed.

REFERENCES

[1] I. F. Akyildiz, D. Pompili, and T. Melodia. Challenges for Efficient Communication in Underwater Acoustic Sensor Networks. *ACM SIGBED Review*, Vol. 1(1), July 2004.

[2] I. F. Akyildiz, D. Pompili, and T. Melodia. Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks (Elsevier)*, 3(3):257–279, March 2005.

[3] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou. Challenges: Building Scalable Mobile Underwater Wireless Sensor Networks for Aquatic Applications. *Special Issue of IEEE Network on Wireless Sensor Networking*, May 2006.

[4] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li. Research Challenges and Applications for Underwater Sensor Networking. In *IEEE Wireless Communications and Networking Conference*, Las Vegas, Nevada, USA, April 2006.

[5] D. B. Kilfoyle and A. B. Baggeroer. The State of the Art in Underwater Acoustic Telemetry. *IEEE Journal of Oceanic Engineering*, OE-25(5):4–27, January 2000.

[6] LinkQuest. http://www.link-quest.com/.

[7] J. Partan, J. Kurose, and B. N. Levine. A Survey of Practical Issues in Underwater Networks. In *Proc. ACM International Workshop on UnderWater Networks (WUWNet)*, pages 17–24, September 2006.

[8] J. Proakis, E.M.Sozer, J. A. Rice, and M.Stojanovic. Shallow Water Acoustic Networks. *IEEE Communications Magazines*, pages 114–119, November 2001.

[9] G. G. Xie and J. Gibson. A Networking Protocol for Underwater Acoustic Networks. In *Technical Report TR-CS-00-02*, Department of Computer Science, Naval Postgraduate School, December 2000.

[10] P. Xie, J.-H. Cui, and L. Lao. VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks. Technical report, UCONN CSE Technical Report: UbiNet-TR05-03 (BECAT/CSE-TR-05-6), Feb. 2005.

[11] P. Xie, J.-H. Cui, and L. Lao. VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks. In *IFIP Networking*, Coimbra, Portugal, May 15 - 19 2006.