2009 IEEE 34th Conference on Local Computer Networks (LCN 2009)
Zürich, Switzerland; 20-23 October 2009

Robust Grid-based Deployment Schemes for Underwater Optical Sensor Networks

Abdullah Reza
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada
Email: areza@cs.ualberta.ca

Janelle Harms
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada
Email: harms@cs.ualberta.ca

***Abstract*—Underwater sensor networks have received significant attention from the research community in recent years. Since radio signals face excessive absorption in the underwater environment, acoustic communication has been the dominant physical layer medium in the literature. Although acoustic communication has long range and omni-directional characteristics like terrestrial radio, it suffers from excessive propagation delay in water and very low bandwidth. In this paper, we consider the design of an optical underwater sensor network based on low cost LEDs and photodiodes. Such an optical communication system has shorter range compared to acoustic systems but is cheaper and can support significantly higher bandwidth. Optical communication requires line of sight which makes optical links vulnerable to occasional failures due to underwater organisms and moving particles. We consider a grid based deployment of underwater sensor nodes and the selection of a topology based on point-to-point optical links that is robust to occasional link failures. We develop patterns for networks with at most 3 interfaces per node constraints. We evaluate the robustness of our proposed deployment patterns by simulating three resilient routing protocols on these patterns and demonstrate that our patterns support a high degree of robustness even though they use only a fraction of all potential links in the grid graph.**

***Index Terms*—Undewater Sensor Networks, Sensor Network Deployment, Optical Communication, Robustness, Fault-tolerance, Topology, Connectivity**

I. Introduction

As a new and evolving field, underwater wireless sensor networks (UWSN) have received a lot of attention from the research community in recent years. UWSNs have immense potential because such networks allow us to monitor the underwater environment which constitutes 70% of the earth's surface. There are many challenges to designing wireless sensor networks due to the harsh underwater environment. Radio and most other electromagnetic signals are excessively absorbed by water, and, therefore, acoustic communication has been considered almost exclusively. Research on medium access control and routing can be found in [11], [13], [22] and in [13]–[15], [18], [23] respectively. While acoustic signals have a range of several hundred meters [1], [2], [4], they suffer from low bandwidth of nearly 5 Kbps [20], very long propagation delay of nearly 1500m/s [14], high and unpredictable error rate [18], [23] and high cost of acoustic modems [20]. The long propagation delay and low bandwidth makes acoustic communication unsuitable for applications that have high bandwidth and short delay requirements, such as real-time and multimedia sensing applications.

Underwater point to point optical communication using LEDs and photodiodes has also been considered in the literature of UWSN. Vasilescu *et al.* [20] experiment with a hybrid UWSN where nodes have both acoustic and optical transceivers. Acoustic communication is used by the nodes to perform localization. An autonomous underwater vehicle (AUV) periodically visits each node and downloads its sensed data using short-range and high-speed optical communication. The delay in data delivery due to the slow physical movement of the AUV makes such a system unsuitable for real-time applications. The oceanographic contamination detection system proposed by Kedar *et al.* [6], on the other hand, uses only directional optical media to communicate under water and a static sink instead of a mobile AUV. Because of the short range of underwater optical communication, they propose the deployment of a number of sensor nodes within the communication range of the sink where all sensor nodes transmit directly to the sink. In order to extend the area of observation, the authors propose the use of multiple clusters with inter-cluster sink communication. A wireless sensor network does not have to stand on its own, that is, it could be attached to a wired network. An example of a wired, sink-based sensor network that has real-time applications is the VENUS project [19] where a sink node is placed on the seafloor connected to the shore using a fibre optic cable. Sensors are connected to this node with wires including cameras to transmit real time information to the shore. The development of a wireless sensor network could be used to extend networks such as this.

Underwater point to point optical networks have properties of higher bit rates, shorter communication range and the need to anchor nodes to ensure line of sight. Vasilescu *et al.* [20] experiment with their underwater optical system with a bit rate of 320 Kbps whereas Giles *et al.* [2] use a mathematical model of a commercially available LED and photodiode to produce a bit rate of up to 4.4 Mbps. There is, however, a reduction in communication range. Depending on the clarity of water, the transmission power, presence of concentrator lenses at the transmitter and amplifiers at the receiver, the maximum communication range varies from 8m to 40m [2], [20]. The range of optical communication decreases as the

water becomes turbid. However, ocean areas that are not very close to the coastal areas usually have clarity very close to pure water [17] allowing larger communication ranges to be supported. Note that wireless optical communication, Free Space Optical, has been proposed in terrestrial sensor, ad hoc and mesh networks [3], [5], [10], [12] but the range of communication is much longer resulting in different design objectives.

In this paper, we consider underwater optical communication using LEDs and photodiodes operating at green/blue visible light as transmitters and receivers, respectively. Additional concentrator lenses and amplifiers can be used with LEDs and photodiodes, respectively, to improve the range of communication. Three design goals are apparent for deploying sensor nodes. First, unlike radio or acoustic communication which have long and omnidirectional transmission range, underwater optical communication has short and directional range. Therefore, connectivity is not inherent in such networks which calls for an intelligent deployment that ensures desired levels of connectivity. Second, optical communication relies on the line of sight property which means that optical links may go down when obstacles are present. Optical links may also go down if a node becomes misaligned. As a result, there should be sufficient redundancy in the topology so that packet delivery can continue even when one or more links are inactive. Third, since transmissions and receptions are directional, if a node wants to communicate in multiple directions, it needs multiple optical transceivers which incurs extra cost. A node could instead use a single transceiver and rotate it in different directions but that would incur extra delay, power consumption and would need extra tuning each time a switch occurs. However, if anchoring nodes is difficult in the environment, this extra cost may be necessary. In this paper, we consider the problem of deploying nodes using multiple fixed transceivers that is robust to temporary obstructions to line of sight. By using alternatives paths, these schemes can also adapt if some nodes become misaligned as well.

II. Problem Description

We consider a 2D grid-based robust deployment of sensor nodes that uses underwater point-to-point optical communication. A grid is a useful topology in sensor networks because it provides comprehensive sensor coverage. All the unit squares have a side of length r where r is the maximum communication range of the optical communication system. Such a communication system can have a value of r ranging from 8 meters to 30 meters depending on the specific communication devices used and the clarity of water [2], [20]. However, our design is independent of the actual value of r and is thus applicable to any configuration.

We assume that each grid point has one sensor node. Each internal node has four 'potential' point-to-point optical links to its four neighbors in the grid, corner nodes have 2 potential links and other edge nodes have 3 potential links. The sink can be either one of the grid nodes or it can be a separate node placed within the vicinity of the grid area. The problem considered in this paper is to select a set of 'actual' links from the pool of potential links to create a connected topology spanning all sensor nodes and the sink with the following three properties:

1) Robustness: Two metrics are considered:

a) Deterministic Robustness: The topology should be 2-edge-connected [21], *i.e.*, the topology should be connected even if we delete an arbitrary link.

b) Probabilistic Robustness: The topology should be such that if we delete an arbitrary number of arbitrary links, a maximum percentage of nodes are still connected to the sink.

2) Path-quality: The topology should have minimum hop paths for each node to and from the sink in order to reduce the energy consumption resulting from communication. The alternative paths in the topology should also be short.

3) Interface-count: Each link in the topology represents one communication interface/transceiver on the nodes at both ends of the link. Therefore, having more links in the topology to achieve the above two properties will introduce more per node and total interfaces in the network, increasing total cost. Thus, we must tradeoff properties 1 and 2 with per node and total interface count in the network.

Our goal in this paper is to select a subset of edges from all the potential edges in the grid so that the robustness and path-quality of the resulting topology reaches a desired level and, at the same time, no node in the topology has a degree greater than a threshold and the total number of edges selected in the process is minimized. We consider cases where each node in the topology is constrained to have no more than 2 and 3 interfaces/degree. We consider placing one node on each grid point and then selecting the direction(s) in which a node can communicate so that when the network starts operating, good paths are available for low-cost communication and alternate paths are available if one/more links are down.

Optical Interface Model: An optical interface consists of a transmitter and a receiver. It is cost-effective and intuitive to place both transmitter and receiver on one interface/board. A node can use one interface for transmitting to and receiving from the same direction. Thus we can model one interface as one undirected edge between two nodes. The deployment task of placing nodes on grid points and setting their interface directions underwater can be done manually or using autonomous underwater vehicles (AUV) [20].

III. Robust Grid-based Deployment Schemes

In this section, we design robust deployment topologies for cases with 2 and 3 interfaces per node constraints.

A. Maximum 2 interfaces per node

We can build a 2-edge-connected topology with 2-degree constraint by forming a cycle spanning all nodes [10] including the sink at a grid point. Since the graph is undirected, there are two paths between any node and the sink. A Hamiltonian cycle can always be formed in a grid graph if either the number of rows or columns or both are even [16]. To improve the

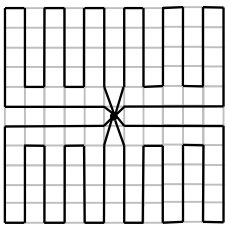


Fig. 1. TOP2: 4 undirected Hamiltonian cycles

design we divide the grid into four quadrants around the sink and form a cycle for each quadrant as shown in Fig. 1. The sink is placed at the center of the grid to minimize average path length. For this design, we assume the sink uses powerful concentrators and amplifiers to increase its range and can hold more interfaces. However, this may require some of the sink's neighbor nodes to also increase their range in order to use alternative paths. This topology is called TOP2.

B. Maximum 3 interfaces per node

With a maximum of 3 interfaces per node, our approach is to first build a 3-degree-constrained shortest path tree from the sink spanning all nodes and then add additional edges to this tree to make it 2-edge-connected and then to further increase the redundancy of available paths to improve probabilistic redundancy. Our target is to minimize the number of 3-degree nodes as we move through each step. We assume that the sink is one of the grid nodes.

We first form a shortest path tree from the sink to all nodes. Our goal is to select a set of edges from the grid to form a shortest path tree from the sink to all other nodes so that no node in the tree has degree greater than 3 and the number of 3-degree nodes in the tree is minimized. We call such a tree an *optimal pattern* for the grid.

Algorithms for generating a degree-constrained topology from a graph in general have been discussed in the graph theory literature [7], [9]. In our work we find a deployment pattern that is 3-degree constrained, minimizes the number of 3-degree nodes and also produces shortest paths in a grid network of arbitrary size.

For our grid, we call the horizontal and vertical lines through the sink the axes. The four quadrants created by the axes are called Q_1, Q_2, Q_3 and Q_4. The Manhattan Distance [8] between two points (x_1, y_1) and (x_2, y_2) is defined as $|x_1 - x_2| + |y_1 - y_2|$ and is the length of the shortest path in terms of hops between the two points in a grid. Since a path can only have horizontal and vertical edges, any path from one node to another that never goes back in a direction already used is a shortest path in terms of hops between the two nodes. This *Manhattan Distance Property* implies that any shortest path between two nodes in a grid must be on or inside the rectangle formed by drawing vertical and horizontal lines through the two points. Within a quadrant, the shortest path routing stays within the quadrant. On the axis, the shortest path stays on the axis and therefore, any shortest path tree spanning all nodes must include all the edges on the axes.

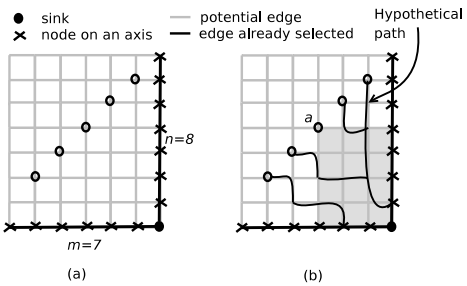


Fig. 2. A quadrant with horizontal axis of length $m = 7$ and vertical axis of length $n = 8$

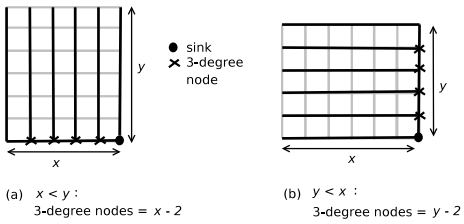


Fig. 3. Optimal pattern for a quadrant.

We now find the lower bound for the number of 3-degree nodes in a 3-degree constrained shortest path spanning tree from the sink for one quadrant.

Theorem 1: Consider a quadrant with sides of size m and n where $m \leq n$. A sink is placed in a corner. A 3-degree constrained shortest-path tree rooted at the sink and spanning all nodes inside and on the boundaries of the quadrant requires at least $(m - 2)$ 3-degree nodes.

Proof: Two axes extend from the sink. Consider the interior nodes of a diagonal from the top (furthest from sink) of the longest axis (the one with size n) to the edge of the quadrant. For example, this diagonal is shown as hollow circles in Fig. 2(a) where $m = 7$ and $n = 8$. There are $(m - 2)$ nodes on the interior of this diagonal (that is, there are m nodes on the diagonal and we exclude the axis node and edge node). We will show that these $(m - 2)$ nodes, call this set of nodes D, require unique 3-degree nodes in their path to the sink.

Suppose we find a shortest path from one node a in D to the sink. According to the Manhattan Distance property, any shortest path between the sink and one of the $(m - 2)$ nodes must be within or on the rectangle created by extending straight lines from the node to the axes. For example, in Fig. 2(b) the rectangle for a is shaded.

To find a shortest path to the sink for this arbitrary interior diagonal node a, two cases can occur. In the first case, the selected path joins a path already created for another node in D. At the first node that it encounters this existing path, it will cause another edge to be added to this node. Note that no other nodes in D are within or on the rectangle of a. Therefore any paths already set up for other nodes of D will create 2-degree or 3-degree nodes within the rectangle (no single degree nodes). Adding an edge to a 2-degree node creates a 3-degree node; adding an edge to a 3-degree node will violate the 3-degree constraint and is not allowed. The other case is that the

path reaches an axis without encountering another path created by other nodes in D. All the edges on the axes have already been selected since these edges cannot be avoided to provide shortest paths to the nodes on the axes. Since the rectangle of a hits an interior axis point/node, this node must be a 2-degree node. Therefore, this also creates a 3-degree node. Both of these cases require the creation of a 3-degree node. Since this is true for an arbitrary node in D, it is true of all $(m-2)$ nodes in D resulting in at least $(m-2)$ 3-degree nodes. □

We now propose an edge selection pattern for a quadrant that results in a 3-degree constrained shortest path spanning tree from the sink with the minimum number of 3-degree nodes. We assume that all the edges on the axes are already selected. We select all the edges that are on the straight line perpendicular to the smaller axis. Then, the nodes on the smaller axis excluding the sink and the far end node have a degree of 3. Thus, the total number of 3-degree nodes is (m - 2) if m is the size of the smaller axis. This is shown in Fig. 3. Note in this figure, that the path selected to each grid node is a shortest path from the sink according to the Manhattan distance property since these paths are formed by never going back in a direction already traversed. Also, no node has a degree greater than 3, thus the pattern indeed produces a 3-degree constrained shortest path spanning tree for the quadrant. In addition, according to Theorem 1, it has the fewest number of 3-degree nodes. Therefore, our pattern is an optimal pattern for a quadrant.

This gives us an optimal pattern for a single quadrant. If the sink is in the middle of the grid and the quadrants are all of the same size, then we can apply this to each quadrant as in Fig. 4(a). According to the Manhattan Distance property, shortest paths to all nodes within or on the boundaries of a quadrant must remain on or within that quadrant, even when we consider the entire grid. Thus, an optimal pattern computed locally for a quadrant remains an optimal pattern for that quadrant when the entire grid is considered, provided that the 3-degree constraint is not violated. If the number of 3-degree nodes in the local optimal patterns for the four quadrants are l_i, $1 \leq i \leq 4$, then their sum $\sum l_i$, $1 \leq i \leq 4$, represents the lower bound for the number of 3-degree nodes in the globally optimal pattern for the entire grid. l_i can be found using Theorem 1. We call this lower bound LB. The case in Fig. 4(a) gives exactly this minimum. However, if the sink is not central causing the quadrants to have different sizes, then the smallest axes may be shared by two different quadrants as shown in Fig. 4(b).

To avoid violating the 3-degree constraint, we must ensure that each quadrant draws its edges from a different (unique) axis. For example, if each quadrant draws its edges from its clockwise right axis around the sink, there is no chance of conflict between two adjacent quadrants. Alternatively choosing the counterclockwise direction would also work. We show in Fig. 5 a pattern that produces a "minimum" number of 3-degree nodes in a quadrant when the chosen axis is the smallest and "minimum+1" when the chosen axis is larger.

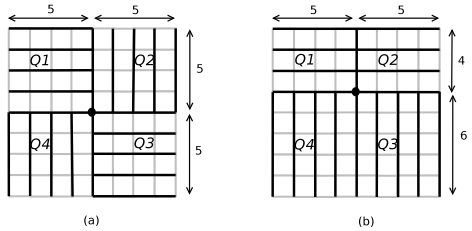


Fig. 4. (a) Optimal patterns for individual quadrants lead to optimal pattern for the entire grid. (b) Optimal patterns for individual quadrants violate the 3-degree constraint.

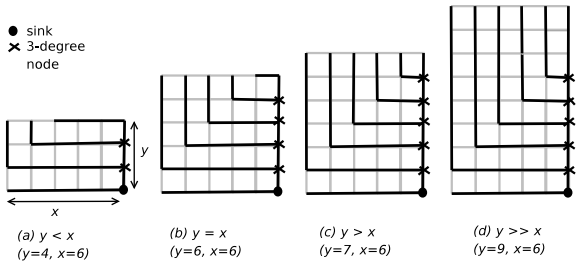


Fig. 5. Pattern using only the vertical axis (clockwise right axis) to place 3-degree nodes

The following corollary describes this for the clockwise case (note that this is easily proved for the counterclockwise case as well).

Corollary: Consider a quadrant Q_i with a clockwise right axis of size y and a left axis of size x. Consider the pattern for Q_i shown in Fig. 5 that draws edges only from the clockwise right axis (y). Fig. 5 shows the pattern for different relationships between x and y. Let the number of 3-degree nodes in an optimal pattern for Q_i be l_i (found using Theorem 1). The number of 3-degree nodes for the pattern in Fig. 5 is l_i if $y \leq x$ or $(l_i + 1)$ if $y > x$.

Proof: For any x and y, the pattern is a shortest path tree (according to Manhattan Distance property) and does not have a node with degree greater than 3. For $y \leq x$, the number of 3-degree nodes is $(y-2)$, as can be seen in Fig. 5(a) and Fig. 5(b). According to Theorem 1, this is the number of 3-degree nodes l_i in an optimal pattern for a quadrant Q_i with $y \leq x$. For $y > x$, the number of 3-degree nodes is $(x-1)$, as can be seen in Fig. 5(c) and Fig. 5(d). According to Theorem 1, the number of 3-degree nodes in an optimal pattern for a quadrant Q_i with $y > x$ is $l_i = (x-2)$. Thus, the proposed pattern has $l_i + 1$ number of 3-degree nodes in Q_i when $y > x$. □

Consider a grid with arbitrary dimension and the sink placed at an arbitrary grid point. If we apply the pattern described in the corollary to individual quadrants of the grid, the overall grid is guaranteed to have nodes of at most degree 3 since each quadrant avoids conflict by placing its 3-degree nodes on its clockwise right axis. For a given sink placement, a pattern can be designed with at most (LB+2) 3-degree nodes where LB (Lower Bound) = $\sum l_i$, $1 \leq i \leq 4$. This can be seen by the following. In general, a pattern created by a clockwise choice of axis will have at most 3 quadrants forced

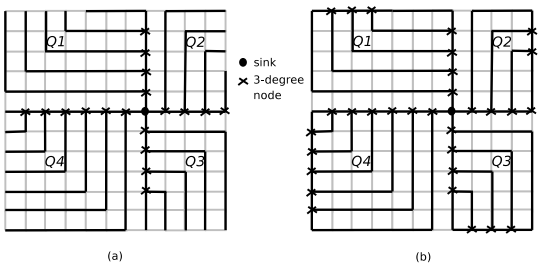


Fig. 6. (a) TOP3: pattern from the corollary applied on 4 quadrants (b) TOP4: Leaves are connected

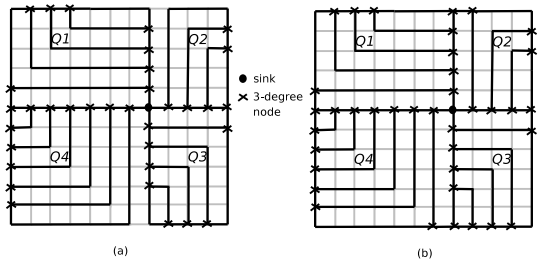


Fig. 7. (a) TOP5: Q_1&Q_4 and Q_2&Q_3 are connected (b) TOP6: Q_1&Q_2 and Q_3&Q_4 are connected

to draw edges from their larger axes, creating (LB+3) 3-degree nodes. However, if this occurs, we could instead choose axes using the counter clockwise direction. In this case, the three quadrants will draw edges from their smaller axes and the number of 3-degree nodes will be (LB+1). There are also sink placements that create (LB+2) 3-degree nodes as shown in Fig. 6(a). Therefore, for any sink placement, we can design a pattern with at most (LB+2) 3-degree nodes. We call this pattern TOP3.

We now add additional edges to TOP3 since it is a tree and will have a low level of robustness. Our first step connects leaves. We note in Fig. 6(a) that the leaves of a quadrant are on the boundary. Thus, we can connect all the leaves of a quadrant as shown in Fig. 6(b) without violating the 3-degree constraint. This is called TOP4 and it has considerably greater robustness than TOP3 since we can now remove any edge from the network, except the 4 edges of the sink, without disconnecting any node from the sink. Thus, TOP4 has high robustness but it is not 2-edge-connected.

We next add two more edges to TOP4 to make it 2-edge-connected and call it TOP5 by connecting Q_1 and Q_4 with one boundary edge and Q_2 and Q_3 with another, as shown in Fig. 7(a). TOP5 introduces 4 more 3-degree nodes. Note that TOP5 is 2-edge-connected since we can remove an arbitrary edge without disconnecting any node from the sink. Our last topology TOP6 adds two more edges, one to connect Q_1 and Q_2 and the other to connect Q_3 and Q_4, increasing the number of possible paths from each node to the sink greatly. This new topology is shown in Fig. 7(b).

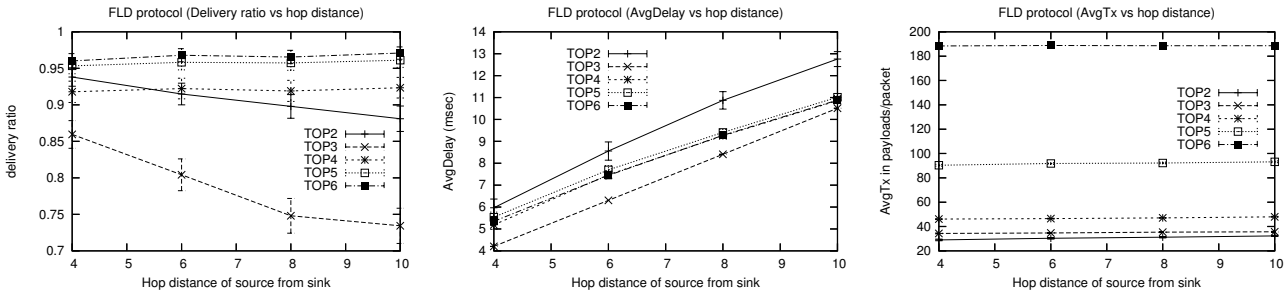
IV. Performance Evaluation

We have designed five topologies for grid-based deployment for different constraints on the number of interfaces a node can have. These topologies are called TOP2, TOP3, TOP4, TOP5 and TOP6 and have been shown in Fig. 1, 6(a), 6(b), 7(a) and 7(b), respectively. A directed variation of TOP2 with 1 interface per node, called TOP1, was also considered but the results are left out since the performance is poor. Table I summarizes some properties of these topologies when applied on a 12x12 grid. As can be seen from the table, our best topology TOP6 has only 165 links compared to 264 links in the potential grid graph. In this section, we simulate routing algorithms over our topologies to measure the robustness of the designs. In our simulation, we use a 12x12 grid with unit distance r = 20m. We place the sink at the center of the grid.

In our simulation, we use 1 Mbps full-duplex links for point to point optical links. We use a single packet generating source node in the network chosen at random at the desired hop distance from the sink. The source node generates ten packets per second, each packet with a payload of 1 Kb. In our simulation, we have assumed infinite buffer space. We use a custom designed simulator written in the C programming language. We run each simulation trial for a simulation time of 20 minutes and take the average of 1200 such trials to generate each data point in our plots. Each plot presented in this section shows 95% confidence interval.

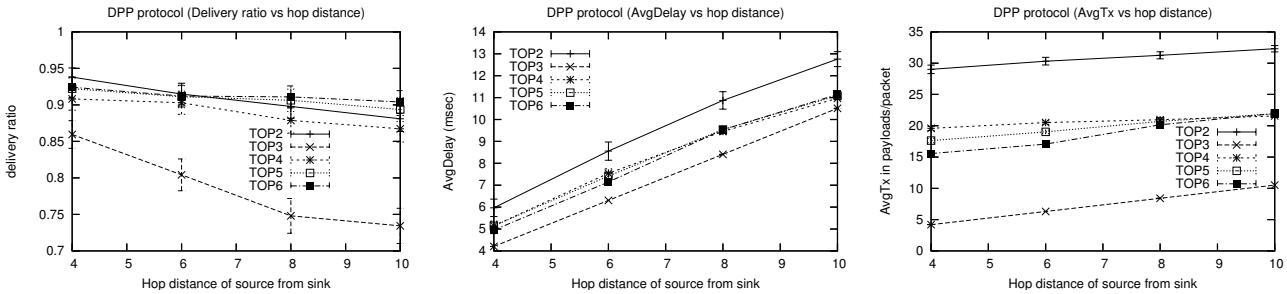
To model underwater obstructions (*e.g.*, underwater organisms, floating objects, sediments), we use ellipses, called *error blobs*, with small semi-minor axis, b, and large semi-major axis, a. At the beginning of network operation, we place three error blobs of dimension a x b at three random locations inside the grid with three random orientations. During network operation, these three error blobs move inside the grid at a constant speed with a random walk pattern. If a link is inside or on the boundary of an ellipse at a particular moment, it is considered down at that moment.

We simulate three simple routing protocols on our topologies since our goal is to evaluate the robustness of the proposed deployment topologies rather than formulating an optimal routing protocol. The first is a simple memory-constrained flooding algorithm called FLD. Although expected to have high overhead in terms of transmission, it will achieve the best robustness. The second algorithm is a simple multi-path algorithm, Dual Paths Protocol (DPP). At set up time, each source selects two node disjoint shortest paths to the sink and these paths are then used to forward packets. TOP3 is a tree so DPP uses the only available path from a source to the sink. With TOP4, the two selected paths go through the same sink edge. The third algorithm is Hop-by-Hop Acknowledgment (HHA) which uses a single path to deliver packets but dynamically adjusts the path to route packets around obstructions. At each hop towards the sink, the receiving node sends an ACK (acknowledgment) packet to the sending node. If the sending node does not receive an ACK before a time limit is passed, the sending node re-computes a new remaining path to the sink taking into account the link failures and forwards the packet on the new path. In order to simplify the protocol, the path information and the link failure information is included in the packet which makes the packet sizes variable. If no path to



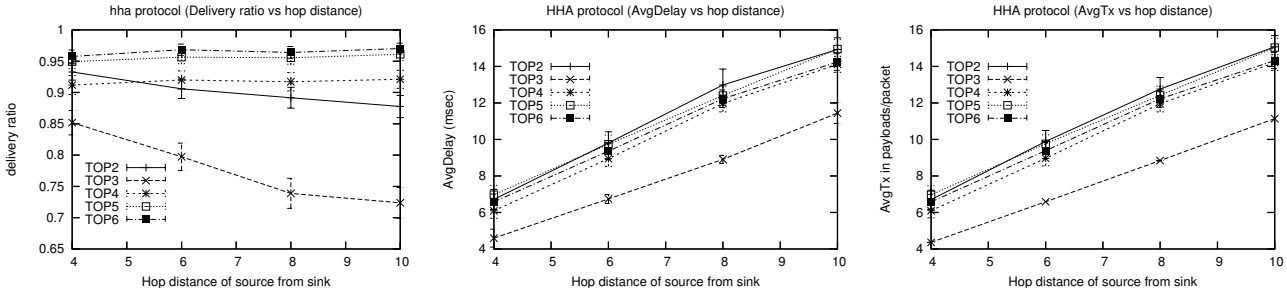
(a) Delivery Ratio (b) AvgDelay (c) AvgTx

Fig. 8. Performance of FLD on different topologies against hop distance (a = 20m, b = 4m and speed = 15 cm/sec)



(a) Delivery Ratio (b) AvgDelay (c) AvgTx

Fig. 9. Performance of DPP on different topologies against hop distance (a = 20m, b = 4m and speed = 15 cm/sec)



(a) Delivery Ratio (b) AvgDelay (c) AvgTx

Fig. 10. Performance of HHA on different topologies against hop distance (a = 20m, b = 4m and speed = 15 cm/sec)

the sink can be found at a node, the node keeps the packet in a queue and tries again after a retry time limit. A packet is dropped if a time equal to TTL (Time To Live) has passed since the generation of the packet. In this simulation, we have assumed that a node can automatically detect the revival of its links (the failure is detected by the lack of ACK packets).

We use three metrics for our evaluation. Delivery ratio is the robustness metric and it is the ratio of the number of packets delivered successfully at the sink to the total number of packets generated at the source. Average Delay Per Packet, AvgDelay, is a measure of the path quality and is the ratio of the total delay faced by all successfully delivered packet to the number of successfully delivered packets. Here, the delay faced by a packet is the time between its generation at the source and its delivery to the sink. Average number of Payloads transmitted per Successful packet, AvgTx, is a measure of overhead and it is the average data transmission (including retransmission, packet overhead and ACK) caused by a successfully delivered packet normalized to the unit of payloads/packet. We also experimented with overhead based on both successful and unsuccessful packets but it has similar relative performance to AvgTx and we do not show the results here.

A. Results

We perform three sets of experiments with each routing protocol applied on our topologies. In the first set of experiments, we vary the hop distance of the source on its shortest path to the sink. We vary the size and speed of the error blobs in the second and third set of experiments.

In our experiments with varying hop distance, we keep the dimension of error blobs fixed at a = 20m and b = 4m and the

speed of error blobs fixed at 15 cm/sec. For HHA, TTL is set to 0.5 seconds. We have experimented with different values of TTL and found that increasing TTL beyond 0.5 sec offers very little improvement in delivery ratio at the cost of unacceptable increase in average delay. A value of 0.5 sec is long enough for packets to travel around error blobs and get delivered but not long enough for error blobs that are blocking all possible paths to the sink to move away in time.

Fig. 8(a) shows delivery ratio of flooding, FLD, on different topologies against hop distance of the source node from the sink. We see that delivery ratio of TOP2 and TOP3 falls steadily with increasing hop distance with TOP2 maintaining higher delivery ratio than TOP3. The probability that a packet faces an error blob increases as the source gets further from sink and the alternative paths are few and long in TOP2 and do not exist in TOP3 which is a tree. Note that with very small hop distance from the sink (4 hops), TOP2 outperforms TOP4 in terms of delivery ratio since in TOP4, all paths from a source node go through a single link (the link connected to the sink) which remains the single point of failure for that quadrant. On the other hand, TOP4 has more flexibility (redundancy) than TOP2 and can handle the sources further from the sink better. TOP4, TOP5 and TOP6 have enough redundancy to keep the delivery ratio almost constant with distance since FLD makes sure that a packet goes through all possible paths from source to sink. We also see a significant increase in delivery ratio as we move from TOP4 to TOP5 since TOP4 has the sink edge point of failure. As we move from TOP5 to TOP6 by adding two more links, the improvement in delivery ratio is small but consistent and visible. With our best topology, TOP6, approximately 97% delivery ratio is maintained with FLD even at a hop distance of 10.

The delivery ratio of Dual Paths Protocol on the different topologies has similar relative performance to FLD, as can be seen in Fig. 9(a). However, there is a decrease in the delivery ratios of TOP4, TOP5 and TOP6 as hop distance increases since longer paths increase the possibility of facing error blobs. HHA protocol has performance very similar to FLD (see Fig. 10(a)), because it also takes advantage of the redundancies inherent in the topology by finding alternative paths.

Fig. 8(b) shows average delay per packet (AvgDelay) of FLD on different topologies against hop distance from the sink. Since TOP3 is a shortest path tree, packets are always delivered on the shortest path from the source to the sink. Hence, TOP3 has the minimum AvgDelay of all topologies and AvgDelay increases linearly with hop distance. TOP2 exhibits the largest and most variable AvgDelay because of the extremely long alternative paths in the cycle. TOP4, TOP5 and TOP6 remain between that of TOP2 and TOP3 and increase steadily with hop distance, although the rate of increase falls with hop distance. The latter observation arises from the fact that the ratio of alternative path length (the path that is used when the shortest path is blocked) to shortest path length is smaller for nodes that are far away from the sink since we add links on network boundaries to increase redundancy. A very important observation is that TOP6, which gives the highest delivery ratio (see Fig. 8(a)), has average delay lower than that of TOP5 and almost the same as that of TOP4. This is due to the fact that since all the quadrants are connected together in TOP6, not only do more paths become available but in many cases better (shorter) alternative paths are found.

DPP has very similar average delay performance compared to FLD since it uses two "shortest" paths from source to sink (see Fig. 9(a)). HHA also exhibits similar relative performance of the topologies but overall higher delays than FLD (see Fig. 10(a)). Since remaining paths are computed at the point of failure rather than at the source, the resulting delivery paths are often considerably longer than the paths selected by FLD to route around error blobs. Table II summarizes the performance of the three protocols for TOP6 at a hop distance of 10.

Fig. 8(c) shows overhead metric, Average number of Payloads Transmitted per Successful packet (AvgTx) for flooding (FLD) on different topologies against hop distance from the sink. None of the topologies demonstrate noticeable change in AvgTx with varying hop distances from the sink since a packet is flooded throughout the network regardless of where the source node is located. The load for FLD is extremely high, though. Although FLD offers high robustness and small delay, it incurs excessive transmissions in the network. With DPP, packets are not flooded anymore, thus we see a huge reduction in AvgTx in Fig. 9(c). Since the path lengths get longer with increased hop distance, we see a steady increase in AvgTx in Fig. 9(c) with higher hop distance for all five topologies. As we move from TOP4 to TOP5 and from TOP5 to TOP6, we see consistent reduction in AvgTx since the length of the alternative path becomes shorter (while the length of the primary path is the same for all three topologies). Overall, unlike FLD, AvgTx is now a function of the path lengths of the primary and the alternative paths instead of the network size. The overhead is further reduced for the HHA protocol in Fig. 10(c). Since HHA delivers each packet on a single path, both delay and number of transmissions are proportional to the length of the path on which a packet is delivered. Consider TOP6, even though HHA uses longer paths in the presence of obstacles, it delivers packets on the shortest path in most cases which causes less transmission overhead compared to FLD and DPP, and supports a delivery ratio similar to FLD.

To see the effect of larger error blobs in the network, we summarize in Table III the performance of FLD, DPP and HHA protocol applied on the best topology TOP6 with the dimension of each error blob increased to (30m x 4m) from (20m x 4m) which we have used in our experiments with hop distance. As can be seen by comparing Table II and Table III, delivery ratio of TOP6 falls with larger blob size with all three protocols although the fall is negligible with FLD and HHA protocols since these two protocols make use of the redundancy inherent in TOP6. HHA supports slightly higher delivery ratio than FLD since with HHA, packets are kept in a queue for some time if no path to the sink is currently available. With HHA, TOP6 supports a very high delivery ratio of 96% even at this very large error blob size. However, HHA faces higher increase in average delay compared to other two

Topology	1-d nodes	2-d nodes	3-d nodes	4-d nodes	Total Edges
TOP2	0	144	0	0	148
TOP3	22	103	18	0	143
TOP4	0	111	32	0	161
TOP5	0	107	36	0	163
TOP6	0	103	40	0	165
Grid Graph	0	4	40	99	264

TABLE I
PROPERTIES OF PROPOSED TOPOLOGIES WHEN APPLIED ON A 12X12 GRID

	FLD	DPP	HHA
Delivery ratio	97%	90%	97%
AvgDelay in msec/packet	10.88	11.14	14.24
AvgTx in payloads/packet	188.60	21.96	14.32

TABLE II
PERFORMANCE OF THREE PROTOCOLS ON TOP6 AT HOP DISTANCE 10 WITH ELLIPSE DIMENSIONS (20X4)

	FLD	DPP	HHA
Delivery ratio	95%	82%	96%
AvgDelay in msec/packet	11.26	11.27	17.40
AvgTx in payloads/packet	184.24	21.41	17.33

TABLE III
PERFORMANCE OF THREE PROTOCOLS ON TOP6 AT HOP DISTANCE 10 WITH ELLIPSE DIMENSIONS (30X4)

protocols since it uses local re-computation of paths which results in longer paths on average in the face of link failures. However, HHA still offers the lowest number of transmissions (AvgTx) since it uses a single path for delivery.

We have also conducted experiments with different speeds of error blobs keeping the size of the error blobs constant. We have found that none of the protocols and none of the topologies show noticeable variation in any of our evaluation metrics since the total area occupied by the blobs in the network remains constant.

V. CONCLUSION

In this paper, we have investigated the design of an optical underwater sensor network. In particular, we have considered sensor nodes deployed in a grid structure and then designed schemes to select point-to-point links between adjacent nodes to generate a robust topology. We have designed a formulation pattern for a 3-degree constrained shortest path tree in a grid rooted at the sink and spanning all nodes in the grid with (LB+2) number of 3-degree nodes in the worst case where LB is the lower bound on the number of 3-degree nodes in such a tree. Our proposed formulation pattern works for any grid dimension and any placement of the sink inside the grid. With this shortest path tree as the base, we have designed a series of deployment topologies that offer higher degrees of robustness by adding additional links on top of the shortest path tree at strategic points. To evaluate the robustness of our proposed deployment topologies, we have simulated three simple routing protocols on our topologies and demonstrated that our best topology TOP6 supports a very high degree of robustness even though it uses only a fraction of edges from the potential grid. Our future work is directed towards proving or disproving that our proposed 3-degree constrained shortest path spanning tree has "minimum" number of 3-dgree nodes and investigating other deployment schemes like hexagonal and 3D grid.

REFERENCES

[1] I.F. Akyildiz, D. Pompili, and T. Melodia. Challenges for efficient communication in underwater acoustic sensor networks. *ACM Mobile Computing and Communication Review*, July 2007.

[2] J.W. Giles and I.N. Bankman. Underwater optical communications systems part 2: basic design considerations. In *Proc. of IEEE MILCOM 2005*, Atlantic City, NJ, October 2005.

[3] P.C. Gurumohan and J. Hui. Topology design for free space optical networks. In *Proc. of ICCCN 2003*, Dallas, TX, October 2003.

[4] A.F. Harris III and M. Zorzi. On the design of energy-efficient routing protocols in underwater networks. In *Proc. of IEEE SECON 2007*, San Diego, CA, June 2007.

[5] A. Kashyap, K. Lee, M. Kalantari, S. Khuller, and M. Shayman. Integrated topology control and routing in wireless optical mesh networks. *Computer Networks Journal*, 51:4237–4251, October 2007.

[6] D. Kedar and S. Arnon. A distributed sensor system for detection of contaminants in the ocean. *SPIE (Society of Photographic Instrumentation Engineers) Vol. 6399, 639903*, Stockholm, Sweden, 2006.

[7] S. Khuller, K. Lee, and M. Shayman. On degree constrained shortest paths. In *European Symposium on Algorithms*, Eivissa, Spain, Oct 2005.

[8] E.F. Krause. *Taxicab Geometry*. Dover, NY, 1987.

[9] M. Krishnamoorthy, A.T. Ernst, and Y.M. Sharaiha. Comparison of algorithms for the degree constrained minimum spanning tree. *Journal of Heuristics*, 7(6), November 2001.

[10] J. Llorca, A. Desai, and S. Milner. Obscuration minimization in dynamic free space optical networks through topology control. In *Proc. of IEEE MILCOM 2004*, Monterey, CA, November 2004.

[11] M. Molins and M. Stojanovic. Slotted fama: a mac protocol for underwater acoustic networks. In *Proc. of IEEE OCEANS'06*, Boston, MA, September 2006.

[12] U.N. Okorafor and D. Kundur. Efficient routing protocols for a free space optical sensor network. In *Proc. of IEEE MASS 2005*, Washington D.C., November 2005.

[13] D. Pompili and I.F. Akyildiz. Overview of networking protocols for underwater wireless communications. *IEEE Communications Magazine*, January 2009.

[14] D. Pompili, T. Melodia, and I.F. Akyildiz. Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks. In *MobiCom'06*, Los Angeles, CA, September 2006.

[15] W.K.G. Seah and H.P. Tan. Multipath virtual sink architecture for wireless sensor networks in harsh environments. In *Proc. of InterSense'06*, Nice, France, May 2006.

[16] S. Skeina. *Implementing Discrete Mathematics: combinatorics and Graph Theory with Mathematica*. Addison-Wesley, Redwood City, CA, 1990.

[17] J. H. Smart. Underwater optical communications systems part 1: variability of water optical parameters. In *Proc. of IEEE MILCOM 2005*, Atlantic City, NJ, October 2005.

[18] P. Sun, W.K.G. Seah, and P.W.Q. Lee. Efficient data delivery with packet cloning for underwater sensor networks. In *Proc. of UT07 and SSC07*, Tokyo, Japan, April 2007.

[19] Victoria Experimental Network Under the Sea. http://www.venus.uvic.ca/.

[20] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke. Data collection, storage and retrieval with an underwater sensor network. In *ACM SenSys'05*, San Diego, CA, November 2005.

[21] D.B. West. *Introduction to Graph Theory, 2nd Edition*. Prentice Hall, New Jersey, 2006.

[22] P. Xie and J.H. Cui. R-mac: An energy-efficient mac protocol for underwater sensor networks. In *Proc. of WASA 2007*, Chicago, IL, August 2007.

[23] P. Xie, J.H. Cui, and L. Lao. Vbf: vector-based forwarding protocol for underwater sensor networks. *UCONN Technical Report UbiNet-TR05-03*, February 2005.