Adaptive Data Delivery for Underwater Sensor Networks

Peng Sun   Winston KG Seah
{sunpeng, winston}@i2r.a-star.edu.sg
Networking Protocols Department, Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613

Abstract- **With the rapid development of terrestrial wireless sensor network technology, and considering that more than 70% of the Earth's surface is covered by water, the increasing research focus on Underwater Wireless Sensor Networks (UWSNs) is not unexpected. Acoustic communications, which is the current viable transmission technique adopted by UWSNs, has a signal propagation delay that is five orders of magnitude slower than radio frequency (RF), and this has a major impact on protocols designed for RF networks. Moreover, the acoustic channel is prone to regional and unpredictable disruptions, resulting in temporal disconnections which can lead to excessive re-routing for conventional routing protocols. In this paper, we develop a data delivery scheme based on a novel multi-sink sensor network architecture with the goal of achieving fast and reliable data delivery in the harsh conditions presented by the acoustic channel. The scheme is designed to dynamically redirect packets when temporal link failures are encountered without requiring network state information to be updated. Using simulations, we showed that the scheme achieves robust and timely data delivery.**

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have been envisioned to trigger revolutionary developments in industry solutions, scientific studies and even human life. WSNs are typically characterized by a number of desirable features, including low cost, short range, multi-hop, fine-grained coverage, and easy deployment. With the fast development of terrestrial WSNs, increasing attention is being focused on underwater wireless sensor works (UWSNs) considering that more than 70% of the Earth's surface is covered by water. UWSNs have promising applications that have yet to be exploited, such as seismic monitoring, offshore drilling, ocean exploration, fishing, sunken wreck recovery, etc. However, there are quite a few research challenges to be addressed before we can effectively apply existing WSN techniques underwater. One of the key challenges is the fundamental difference in the physical channel.

Due to the high attenuation of RF signals, underwater networks employ acoustic communications as the physical transmission method. Acoustic signals travel at the speed of sound (1.5×10^3m/s) instead of light (3×10^8m/s), which is five orders of magnitude slower than the terrestrial RF signals. While the propagation delay is negligible for terrestrial RF-based WSNs, it cannot be ignored for UWSNs. Besides the propagation delay, the harsh environment in the oceans also leads to more intermittent packet loss over acoustic links. Link quality also fluctuates with temperature, depth, water currents, and ambient noise. Worse, this type of disruption is hard to predict. Yet another peculiarity is the occurrence of regional 'blackouts'. This type of link disruption tends to be spatially clustered, and it is mostly caused by sudden change of water current or moving objects like schools of fish. Such disruptions result in regional link failures, rendering traditional data delivery protocols temporarily inoperable. After the event has past, the link may automatically recover. Unfortunately, such event durations are unpredictable and often too long for typical logical link control protocols to handle.

In this paper, we further enhance the multipath virtual sink sensor network architecture [1][2] with an adaptive data delivery scheme, in order to achieve efficient, timely, and reliable data delivery over lossy links with long propagation delay and spatial temporal blackouts. This paper is organized as follows. In section II, we define our problem and scope using typical applications. In section III, we present our multi-sink data delivery scheme and evaluate the performance in Section IV. Section V discusses related work, and lastly, we conclude and discuss future work in section VI.

II. PROBLEM FORMULATION

Sensor networks are primarily application driven. We take a typical scenario, offshore drilling in deep waters (shown in Figure 1) as the application for an underwater sensor network and use it to define the problem and scope.

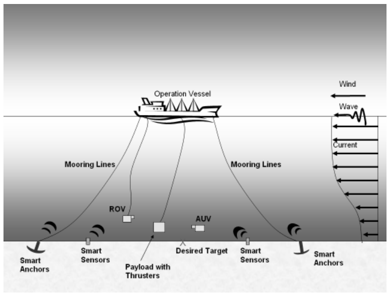


Figure 1. An Offshore Drilling Scenario

Offshore drilling in deep oceans typically employs a floating rig or operation vessel, held in place by cables fixed to smart anchors on the sea-bed. The distance between the smart anchors can be 5-10kms depending on the depth. Within the area encompassed by these anchors, smart sensors are deployed for monitoring, navigation of undersea autonomous vehicles, assisting surface operations or other purposes. Sensors are deployed with inter-node distances ranging from 300m to 500m, and data produced by the sensors need to be transmitted to the surface promptly. Without loss of generality, Figure 2 illustrates our model of the underwater sensor network topology with four sinks at the four anchors of a floating platform.

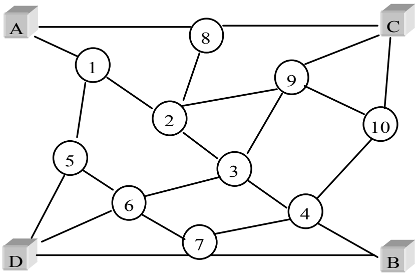


Figure 2. Four Sinks with Ten Sensors

Data are being delivered to any of the four sinks, and four sinks are linked to the surface via high-speed links, enabling them to share their network knowledge 'instantly', forming one virtual sink. We target applications with the following features: 1) Data are generated by sensors to be delivered to the sinks periodically; 2) Sensors are anchored and generally static; any motion due to currents is expected to be small; 3) Sensor nodes are homogeneous; and 4) Density of nodes is sufficient to cover the whole region.

A. General Application Assumptions

We target applications with the following features: 1) Data are generated by sensors to be delivered to the sinks periodically; 2) Sensors are generally static; any motion due to currents is expected to be small; 3) Sensor nodes are homogeneous; and 4) Density of nodes is sufficient to cover the whole region.

B. Goals and Challenges

The goal is to achieve efficient, timely, and reliable packet delivery over lossy links with long propagation delay and spatial temporal 'blackouts'.

The propagation delay, which is five orders of magnitude higher than RF, makes automatic repeat request (ARQ) techniques that are commonly used in terrestrial networks for packet loss detection very inefficient. Furthermore, error recovery methods like retransmission incur excessive latency and signaling overheads. Forward error correction (FEC) techniques appears to be more suitable for providing robustness against errors but this incurs additional overheads that compete for the already scarce bandwidth; the processing needed further drains the critical energy resources. It has thus been suggested that the long propagation delays are better addressed at the network layer [4].

Underwater links typically have much higher bit error rates than terrestrial RF links. This can be addressed from different layers of the protocol stack, e.g. robust error-correction-coding and modulation schemes at physical layer, code division multiple access at the data link layer, etc. By assuming that link layer retransmission schemes are not employed, we model the lossy link with a per-hop loss-ratio (PLR), $0 \leqslant PLR \leqslant 1$. High PLR means a bad link, and low PLR means a good link. Similarly, a better physical layer modulation scheme can be reflected as a better PLR.

The last challenge is regional temporal 'blackout' or loss of connectivity, which we model with temporary PLR=0 on a group of links. Each 'blackout' will happen for certain period, and after that period, the links will revert back to their original PLR values.

III. MULTIPLE-SINK APPROACH

In typical single sink wireless sensor networks, data packets converging towards the sink results in channel contention and rapid energy depletion of nodes near the sink. By deploying a set of physically diverse sinks, traffic can be spread among the sinks and also away from regions of temporal blackouts that are common in underwater acoustic channels. The multipath (multiple) virtual sink approach using a simple reverse path routing algorithm has been shown to be effective for underwater networks [1]. Here, we propose an adaptive multi-sink data delivery scheme that will further exploit the features of this network architecture.

A. Gradients for Data Forwarding

A sensor node's gradient is then defined as a collection of minimum hop counts from all available sinks. A gradient, the shortest logical distance, should not change frequently once it is established. In Figure 2, where there are 10 sensors and four sinks, A, B, C, and D, each sensor will have four gradient values corresponding to the fours sinks, as listed in Table 1. For example, node 10 can be reached by sink C either directly (1 hop) or through node 9 (2 hops), and the gradient value is 1, instead of 2.

TABLE 1. Gradient Values

Sensor	A	B	C	D
1	1	4	3	2
2	2	3	2	3
3	3	2	2	2
4	4	1	2	2
5	2	3	4	1
6	3	2	3	1
7	4	1	3	1
8	1	4	1	8
9	3	3	1	3
10	4	2	1	3

A.1 Gradient Setup

The gradients are setup immediately after the network is deployed by letting each sink disseminate hop-count messages. A node *i*'s gradient is a collection of numbers, denoted as $G(i) = (g_{x_1}, g_{x_2} \cdots, g_{x_n})$ where g_{x_i}, $1 \leq i \leq n$, represents the minimum hop count for sink x_i. Gradient setup is triggered by a broadcast message from each sink during the network setup and initialization phase, and relayed through the network, until every node has established its gradients. At very beginning, each node i sets its gradients to ∞ for all sinks, i.e. $G(i) = (\infty, \infty, ..., \infty)$. The broadcast message from each sink has a gradient value of 0, that is, g_A=0, g_B=0, g_C=0, and g_D=0 for sink A, B, C, and D respectively. For every node, when it receives a broadcast message from its neighbor with a gradient value g_x, which represents the neighbor's gradient value for sink x, it will compare g_x+1 against its own value for sink x, and the smaller value is chosen as its new gradient. If a node has updated its gradients, it will also broadcast a message with its updated gradient values.

To mitigate a potential broadcast storm, a node will wait an arbitrary short duration before it relays/rebroadcasts the message [6]. This also allows it to receive as many neighbors' messages as possible, and thus reduces the number of rebroadcasts. This waiting period can depend on how far a node is from the sink, e.g. the further it is from a sink the higher the probability of it receiving more broadcast packets, and therefore it waits longer. This allows it to hear as many of its upstream neighbors' broadcasts as possible, and only needs to broadcast its gradients once. Alternatively, nodes can also combine gradients from different sinks into one message. These methods can significantly reduce the total number of broadcast messages, as shown later in Section IV.

A.2 Source to Sink

Packet forwarding from source to sink is not difficult when each link is in good condition. To forward a packet towards a particular sink, a forwarding node selects a neighbor node that has a gradient towards that sink. For example, when node 6 wants to deliver a packet to sink C, it needs to forward the packet to its neighbor 3, because node 3 has a smaller gradient for sink C. The path from 6 to C is 6→3→9→C. When there is more than one candidate, it can select one, some, or all of them. For example, node 2 can select node 1, 8, or both to send to sink A. For our performance studies, we arbitrarily select one path to forward the data. However, this selection process deserves further studies.

A.3 Sink to Source

Data flow in a sensor network is typically asymmetric from the source (sensor) to the sink. When the sink needs to query the sensors, the query message is usually flooded through the network. A query to a specific sensor can simply take the reverse data path from the sensor to the sink. Alternatively, we first identify a source node by its ***id*** and gradient; e.g., node 2 will be identified as (2, 2, 3, 2, 3). Then, using Table 1, from sink x to node 2, a next hop node is selected by satisfying two conditions: 1) the gradient for x increases toward that of destination, and/or 2) the overall gradient difference decreases. The first condition ensures that the packet goes towards the source while the second condition ensures that the logical distance decreases. If more than one next hop satisfies the two conditions, the largest value[1] which translates to maximum progress is selected. Moreover, we ensure each hop is always approaching the destined source, and the route is loop free.

Table 2 illustrates how a path from sink D to node 2 is found. The first hop is picked from one of sink D's neighbors, i.e. nodes 5, 6, and 7. Firstly, each increases the D-gradient by one, meaning they are moving away from D (condition 1.) To get the gradient difference, we first compare the gradient of the destination (node 2) against the gradients of these three neighbors, and then add all the values to obtain the total gradient difference. Node 5 with the smallest difference is selected for the first hop. The second hop starts from node 5, which has two neighbors, viz. nodes 1 and 6. Node 1 increases the D-gradient by one, while node 6 does not; thus node 1 is selected for the second hop. Lastly, the next hop has only one candidate which is the destination, node 2. Finally, the zero gradient difference and the node id are then used to identify the destination node.

TABLE 2. Path Finding Process

Hop	From	To	Gradient values	Gradient diff. between node & destined source	Select
1st	D	5	(2,3,4,1)	(0,0,2,2) → 4	✓
		6	(3,2,3,1)	(1,1,1,2) → 5	
		7	(4,1,3,1)	(2,2,1,2) → 7	
2nd	5	1	(1,4,3,2)	(1,1,1,2) → 4	✓
		6	(3,2,3,1)	(1,1,1,2) → 5	
3rd	1	2	(2,3,2,3)	(0,0,0,0) → 0	✓

A.4 Handling Link Quality Fluctuations and Failures

In underwater sensor networks with static and quasi-static nodes, while the quality of a link connecting two nodes can fluctuate substantially and in the worst case make the link dysfunctional for periods of time, the ability to transmit over that link remains as long as the two ends (nodes) of the link are still operating. Hence, we can assume that the minimum hop count from each sensor to each sink remains unchanged unless the network topology has been permanently altered. For example, some sensor nodes have permanently failed or their links have been physically impaired, e.g. by objects falling between them and severely blocking the transmission paths causing the network topology to be permanently altered.

When there is link failure, the gradient can no longer accurately reflect the network topology, and the data delivery scheme may not be able to find a path to deliver the data. For example, when the link between node 10 and sink C is broken, node 4 cannot find a route to C. A typical and immediate remedy that comes to mind would be to initiate a gradient update. However, we argue that it is infeasible for UWSN due to two reasons. First, a gradient update may cause a chain of updates by more than one node, for example, when link 1→A is broken, the gradient of node 5 will be affected, and in turn node 6 will be affected. This chain effect is harmful as essentially propagates local information to entire network, which is unnecessary and limits the scalability. Secondly, updating gradients dynamically can easily result in routing loops, due to the temporal inconsistency of the gradients and

---

[1] The largest value may be shared by more than one node, and we arbitrarily select one, as in Section *A.2*.

coupled with the long propagation delay, this state inconsistency can persist for a long time.

The aim is to avoid letting transient topology changes arising from link quality fluctuations initiate repeated unnecessary gradient updates. Hence, we propose not to initiate local the gradient updates but address the problem using another packet delivery method, namely, multi-sink multi-path delivery (cf: Section III.*B*.) On the other hand, gradient updates are necessary when there are permanent network topology changes and this can be achieved by scheduling network management and maintenance phases.

B. Multi-Path Data Delivery

Typical multipath routing protocols setup multiple routes between a pair of communicating nodes [11]. Depending on how the routes are selected, there is a strong likelihood of contention occurring among nodes that are on different routes but are close to one another. The contention becomes more serious when different path converge at the destination. This phenomenon may be somewhat reduced if the transmissions over different paths are not done simultaneously but this inadvertently increases the overall transmission latency. Moreover, the longer a transmission takes, the more likely it is to be subjected to link quality fluctuations. Besides contention, if the link condition in the sink's vicinity is poor, the multipath transmissions towards that sink are unlikely to be successful.

Our multipath routing scheme is built on the multi-sink architecture and delivers the packet to spatially diverse sinks. There is no converging point for different paths, and thus it eliminates that near-sink contention problem. It also avoids sinks that are experiencing poor link conditions.

Multipath routing can also be viewed as 'simultaneous retransmission' instead of attempting the retransmissions sequentially, and thus it reduces the overall packet delivery delay besides improving the chances of data delivery. One potential drawback with multipath routing is that it may result in many redundant duplicate packets delivered to the sinks. Hence, careful path selection is critical. Intuitively, when the overall link quality improves, we should use fewer paths to avoid duplicates. Conversely, when link quality degrades, more paths should be used to increase the reliability.

We present and compare two delivery methods: towards a single sink or multiple sinks simultaneously, and then evolving to the optimal path(s) with the adaptive mechanisms.

B.1 Single Sink

A single sink delivery starts by selecting a sink with the best gradient. For example, in Figure 2, node 5 will select sink D since it is the nearest sink according to Table 1. However, should the sink be unreachable due to link failures, the packet will be redirected to another sink. This process requires the ability to detect packet loss as a result of link failure and we utilize a per-hop acknowledgement strategy. End-to-End acknowledgement is not feasible due to the long propagation delay. A node will request for acknowledgement (e.g. flag in the packet) from next hop. After Txn_{Max} transmissions without acknowledgement, where Txn_{Max} is the maximum number of trials, we deem the link as broken. As the long propagation delay of the acoustic channel, we should not wait too long for the ACKs, and thus select $Txn_{Max} = 4$ (see Appendix.)

When a packet is forwarded towards one sink that is unreachable, the network should redirect it to another sink. With our gradient setup, it is easy to redirect the packet to a new sink while it is being forwarded. For example, in Figure 2, when link 5→D is broken, node 5 will 'blacklist' sink D for that particular packet, and select one of the nearest sinks from the remaining three. Since sink A has the smallest gradient value of 2, node 5 will forward the packet towards A, using path 5→1→A. If link 1→A is broken as well, then A will be added to the blacklist, and the packet will be delivered to C, since node 1 sees a smaller gradient value of 3 for C than for B which is 4. If all trials fail, then the packet is dropped. The blacklist method guarantees no loops are formed when attempting to forward to each sink.

B.2 Multiple Sinks

Another approach is to forward a packet to all available sinks as it avoids dynamically changing destinations. Unlike the single sink delivery method, this does not use per-link acknowledgements or retransmissions. It avoids loss detection and packet redirection, and reduces overall delay needed to reach a sink. An adaptive mechanism is necessary to find an optimal balance between cost (duplicates) and success rate. In multiple sink delivery, a source node first tries to send packet to all sinks, expecting feedback (see Section III.*C*) information within a maximum timeout period[2]. If nothing is received despite sending to all sinks, the source node will increase the number of per-hop retransmissions. On the other hand, if the feedback reports too many duplicates, the source will reduce the number of sinks to forward packets to. The sink with the lowest successful delivery rate will be dropped first and the adaptation process continues, adding or dropping sinks until the overall number of duplicates per packet approaches one.

C. Collaborative Feedback

After a packet arrives at one of the sinks, all sinks will be aware of this delivery as they collectively form a virtual sink [1]. They can then feedback to the source node the delivery ratio of each path, and the source node will then be able to select a combination of sinks for subsequent data delivery. Feedback information includes packet arrival rate on each sink, and the number of duplicates.

Feedback can also take advantage of the multi-sink architecture. While a source node delivers packets towards one sink, the feedback packet can be sent from another sink to the source node. Feedback can also be sent simultaneously from multiple sinks, and loss of packets can be used to deduce useful network dynamics. For example, when a node delivers packets to A and B, and feedback comes from C and D, a

---

[2] Time out can be calculated by the gradients and per-hop delay.

missing feedback packet from D will imply a broken path from D, and the feedback will report the delivery status of A and B to the node.

Feedback can also come from an intermediate node if it has acquired information on the delivery status of different sinks. For example, if node 2 and node 6 already know that sinks A and D are unreachable, they can feedback to node 3 when node 3 tries to send packets to these two sinks. However, outdated knowledge in intermediate nodes may result in incorrect feedback. Feedback from the sink(s) can also be sent periodically, and by monitoring the loss of feedback, a source node will also be able to infer the path status for each sink. Both intermediate node feedback and periodic feedback deserve further study on their cost and gains. In this paper, we assume that feedback is sent from all available sinks in our performance studies below.

IV. PERFORMANCE EVALUATION

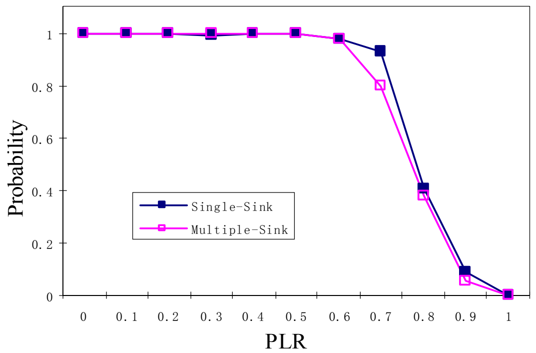
To validate effectiveness and efficiency of our proposed data delivery scheme, we make use of simulations developed on Qualnet [12]. We consider an underwater rectangular region of 2.5km by 2.5km with 200 nodes deployed uniformly in the region. Four sinks are at four corners. Table 3 lists other simulation parameters.

TABLE 3 Simulation Parameters

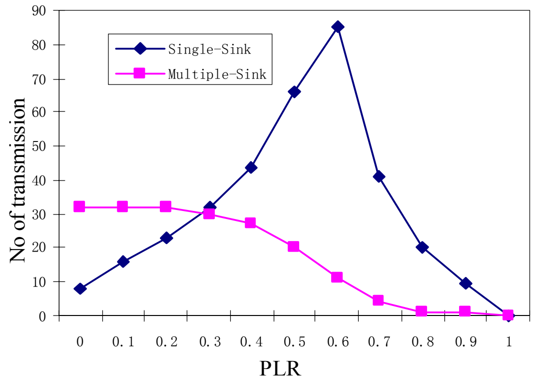
Parameter	**Value**
Transmission Range	250 meters
Channel Frequency	15Khz
Data Rate	5000 bits / second
Speed of Sound	1500 ms^{-1}
Propagation Loss (spherical [6])	20 $\log(R/R_{1m})$ where R=radial distance from the source; and R_{1m}=1m=reference unit distance.
Data traffic	Constant Bit Rate (CBR) with packet size of 256 bytes. All sensor nodes send packets sequentially at time intervals of 100seconds.

A. *Single vs Multiple Sink Delivery*

First, we evaluate the performance of the two delivery methods based on two metrics: 1) probability of a packet successfully reaching a sink (benefits), and 2) average number of transmission needed to reach a sink (costs). In Figure 3a, the success rate of both methods are above 90% before the link quality deteriorates beyond 60%. When the PLR exceeds 60%, the probability of successful delivery for the multiple sink method drops significantly. The single sink method performs better at this level of link quality, due to its retransmission mechanisms. However, due to high packet loss, the retransmission mechanism also incurs a high cost, as shown in Figure 3b. When the link quality deteriorates further, neither method works. A more viable approach would be to buffer the packets temporarily and forward them later when the link quality improves.



a) Probability of Reaching a Sink



b) Number of Transmissions needed to reach a sink

Figure 3. Single vs Multiple Sink

We observed that multiple sink delivery has a better performance in terms of transmission costs, especially when the link quality deteriorates beyond 30% loss. Moreover, it incurs shorter delays because it forwards to all sinks simultaneously instead of waiting and retransmitting. Thus, we adopt the multiple sink delivery method since it is able to achieve comparable performance as the single sink method with lower costs and delays.

B. *Robustness*

Next, we evaluate the average packet delivery ratio (PDR) of the network. PDR is given by the total number of unique packets received by all sinks as a fraction of the total number of packets that are generated by the sources. If both sinks A and B receive a packet with the same sequence number from the same source, the number of unique packets received is counted as one. We average PDR values from all source nodes to give the PDR for the network.

B.1 Lossy Links

In the network, we simulate the unstable acoustic links assigning every hop a PLR value ranging from 0 to 1 where 0 means no packets are lost, and 1 means all packets are lost; i.e. the larger the value, the poorer the link quality. Figure 3a shows our data delivery scheme achieve a high overall success rate when per hop PLR is less than 50%, which is quite desirable for most underwater networks. When the PLR exceeds 60%, the PDR drops rapidly. The performance is attributed to two reasons. Firstly, the paths toward different sinks diverge and do not interfere with one another, and secondly, unless link quality is poor throughout the network, it is unlikely that all sinks are unreachable.

B.2 Regional Blackout

To simulate regional blackouts, we select a square region within the network and mark all links inside the region as broken, i.e. PLR = 0. This blackout region changes its location randomly every 3600 seconds, which is sufficiently long to cause traditional routing protocols to fail. We study how large the blackout area can grow as a fraction of the whole network by varying from 1% to 30%.

Instead of monitoring all nodes in the network, we focus on the sensor node in the center of the network and observe how the data delivery scheme overcomes regional blackouts. We do not allow the region to cover the center node, since all neighboring links from the source node would fail and the PDR will definitely drop to 0, which is exactly what happens when the blackout region goes beyond 25% of the total area. If we choose a node near a corner and the blackout region totally isolates it from the rest of the network, then any data delivery scheme will not function and the best option would be to buffer the packet until the link quality improves.

The almost constant PDR depicted in Figure 4 shows that the source node can always deliver packets to a reachable sink, no matter where the blackout region is, which demonstrates the robustness of our data delivery method against regional link failures. As long as the regional blackout does not isolate or cover a node, it cannot kill all possible paths to all the sinks and our scheme can always find a path to deliver the packets.

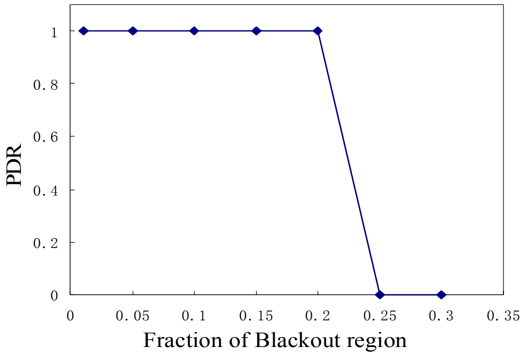


Figure 4. Robustness against regional blackouts

C. *Timeliness*

We define a metric to measure timeliness which we call the timeliness-factor. For a source node, it is obtained as follows. We first measure the average shortest time taken by its packets travel to each sink. Then, we compare it against the delivery optimal time, which is calculated by adding up the sound propagation delay on the shortest path. We then compute the timeliness-factor as the ratio of average packet delivery time to the optimal time. Let us assume that the shortest delay for node 2 to reach sink A is through path $2 \rightarrow 1 \rightarrow A$, and the optimal time for node 2 to deliver a packet is the summation of delay over twos, say 200ms. Similarly, we can obtain optimal time from node-2 to B, C, and D, say 300ms, 200ms, and 300ms respectively. Then we measure the average shortest packet delivery time to A, B, C, and D as 600ms, 700ms, 800ms, and 900ms respectively. The timeliness-factor for node 2 is 2.58, which is computed as shown in Figure 5.

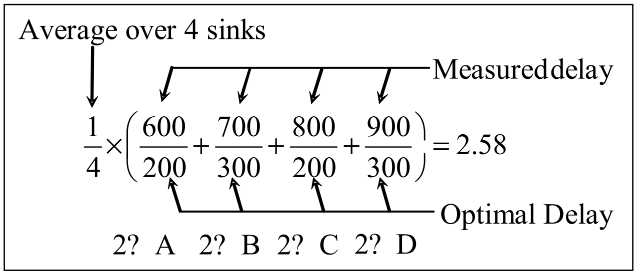


Figure 5. *Timeliness-factor* for single node

We simulated a network with a blackout region of 15%, and measured the timeliness-factor over all possible values of PLR, and the results are shown in Figure 7. The small values show that our data delivery scheme is approaching the optimal delivery time, thus it can be deemed as 'fast'. This is logical as it does not have significant delay on routing path. The delay comes mainly from path selection.

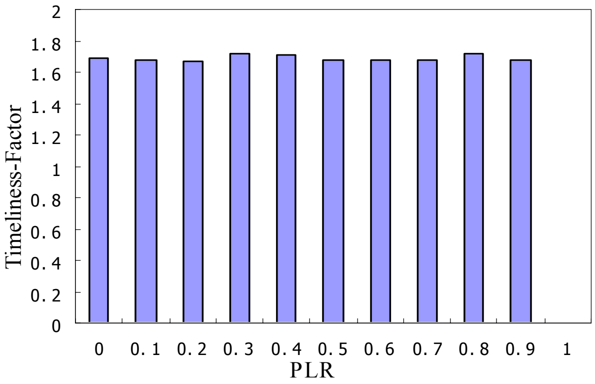


Figure 6. Timeliness

D. *Energy Consumption and Efficiency*

We measure the energy consumption implicitly using the total number of transmissions. As signal attenuation is very high in the underwater acoustic medium, transmission essentially forms the bulk of energy consumption. From Figure 3b, we can deduce that the energy consumption is reduced when packet loss increases. Instead of initiating retransmissions, the scheme actually reduces transmission when PLR goes towards one. The efficiency is kept high, as the scheme does not incur transmission cost to achieve comparable delivery success.

We also measured the packet redundancy to validate the efficiency of our scheme. The redundancy factor is given by the total number of unique packets received at the sinks as a fraction of the total number of packets received. The redundancy factor is always smaller than one and a smaller value implies more wastage of network resources. Figure 7 shows our results. For PLR from 0.1 to 0.9, the overall redundancy is almost one implying almost no redundancy. The slight redundancy mainly occurs during the evolving stage when the scheme has not found the optimal paths yet.

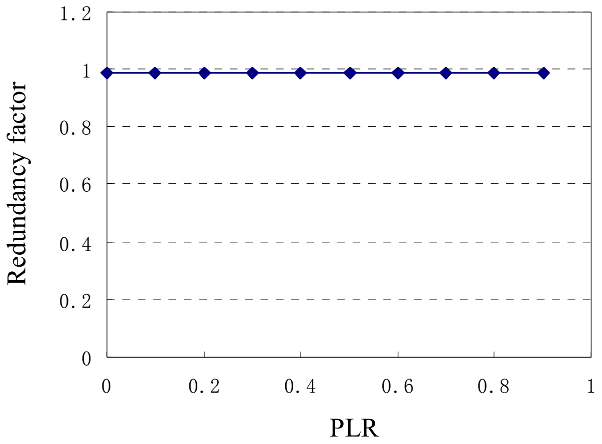


Figure 7. Redundancy Factor

E. Evaluation of Gradient Setup

In Section III.*A*, we discussed the gradient setup procedure performed during networking initialization. To alleviate the number of broadcast messages, we proposed two methods, viz. wait before forwarding gradient setup messages, and combine gradient messages from four sinks. Here, we evaluate and discuss the effectiveness of these two methods.

As expected, the total transmissions during the gradient setup phase decreased as the average waiting time increased. During the waiting period, a node can receive more packets and every incoming packet can potentially change its gradients. Thus, receiving more packets before relaying to its neighbors is effectively reporting its own gradient updates with less frequency. However, waiting for too long does not further reduce the total transmissions as it would have received most, if not all, incoming messages.

By combining the gradient messages, a node can significantly reduce the number of transmission by transmitting one instead of as many packets as the number of sinks. This is also clearly shown in Figure 8. When messages are combined, the total transmissions are significantly lesser and decreases much more rapidly.

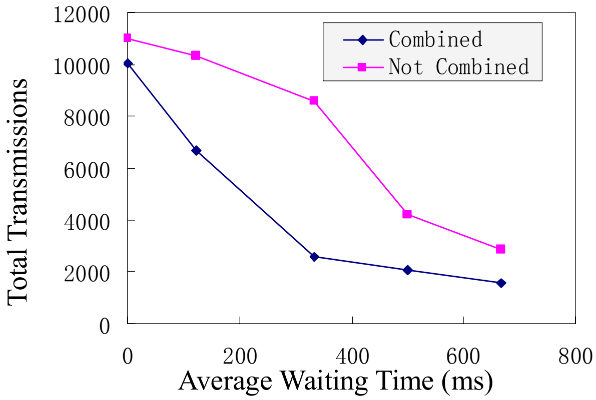


Figure 8. Reducing gradient setup overheads

V. RELATED WORK

With the growing interest in underwater sensor networks, considerable amount of research effort and resources have been put into studying issues like network architectures, medium access control, topology management, routing and data dissemination protocols, energy efficiency and power control, localization and tracking. A wide range of research challenges covering all layers of protocol stack have been discussed [4]. A major focus of research on medium access control (MAC) is on new protocol design to address large propagation delays, rather than to simply adapt existing MAC protocols [3]. Our aim of modeling each link with a per-hop loss ratio is to abstract the features of the lower layers like MAC. Similarly, physical layer modulation schemes and bit error correction [10] will contribute to a better link quality and represented as lower per-hop loss ratios for the links. As new schemes emerge, we will then incorporate the new results into our design.

Another point highlighted is the need to develop routing algorithms that are robust with respect to the intermittent connectivity of acoustic channels [4]. Position-based approach like VBF [8] assumes the availability of 3-D positioning information which may be hard to realize in actual deployment scenarios. Studies have also shown the impact of different type of traffic types, such as delay sensitive and delay insensitive, on the network performance [9]. This is exactly the objective of our proposed scheme. While multiple sink data delivery schemes have been proposed for terrestrial sensor networks [13][14], the characteristics of the terrestrial wireless sensor networks are distinctly different from those of underwater sensor networks [4] and hence, like many existing routing and data delivery schemes for terrestrial wireless sensor networks, cannot be applied. Our approach has been designed specifically to address the salient characteristics of the underwater environment.

VI. CONCLUSIONS

In this paper, we further enhance the multipath virtual sink sensor network architecture [1] with an efficient data delivery scheme. It eliminates the need to measure link status before sending data, which is very desirable for acoustic channels, as the measurement itself may take a long time and is error prone. Our scheme does not rely on any location information, and thus giving it more potential for deployment. The use of collaborative feedback mechanisms enables the scheme to be adaptive, and we can deduce network status from *loss of information*. From the performance evaluations, we confirm that the scheme is able to deliver packets promptly with good reliability and timeliness, even in the presence of regional link blackouts.

An outstanding question remains, which is the optimal number of paths to use for data delivery in order to maximize the delivery ratio while minimizing the communication overheads and redundancy (when the channel conditions are good.) An adaptive scheme is dependent on its ability to sense the environment accurately in order to react appropriately. These, among others, are some of the ongoing efforts to realize viable underwater sensor networks.

APPENDIX

We determine the number Txn_{Max} as follows. Given a lossy link with PLR = p, the number of acknowledgements received with every N transmissions will be $\lfloor\lfloor N \times (1-p)\rfloor \times (1-p)\rfloor$. Figure 9 shows the relationship between various N and p values.

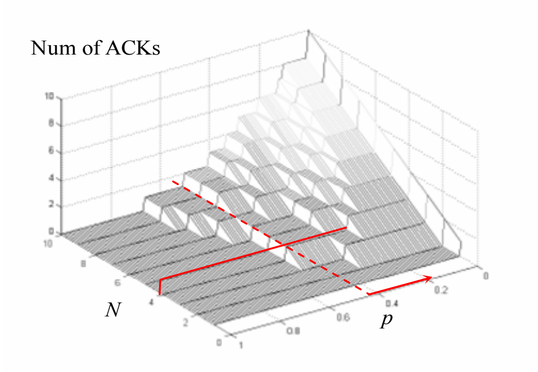


Figure 9. Expected Number of ACKs

We split the N-p plane into two parts, as denoted by the dotted-line (roughly at $p=0.42$) in Figure 9. The first part on the right (marked by the arrow) where $0<p<0.42$ corresponds to the cases that have at least one ACK, while the second part only has a very small portion with at least one ACK. The first part implies less than four trials is required as packets are received and ACKs are generated, while the second part implies that more than four trials are required as packets are lost and therefore no ACK is generated; furthermore, ACKs that are generated may also be lost. As we can see from the figure, when N increases beyond 4, the portion that indicates at least one ACK does not increase much.

REFERENCES

[1] W.K.G. Seah and H.P. Tan, 'Multipath Virtual Sink Architecture for Wireless Sensor Networks in Harsh Environment', invited paper, *Proc. of InterSense 2006*, May 29-31, 2006, Nice, France.

[2] W.K.G. Seah and H.X. Tan, 'Multipath Virtual Sink Architecture for Underwater Sensor Networks', *Proc of the MTS/IEEE OCEANS2006 Asia Pacific Conference*, May 16-19, 2006, Singapore.

[3] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, "Research Challenges and Applications for Underwater Sensor Networking". *Proc of the IEEE Wireless Communications and Network Conference* (WCNC2006), Apr 3-6, 2006, Las Vegas, Nevada, USA.

[4] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges", *Ad Hoc Networks*, Vol 3, No 3, May 2005, pp 257-279.

[5] H. Füβler, et al., "Contention-Based Forwarding for Mobile Ad-Hoc Networks", *Ad-Hoc Networks*, vol. 1, no. 4, pp. 351-369, Nov 2003.

[6] P.C. Etter, *Underwater Acoustic Modeling and Simulation*, 3rd edition, Spon Press, New York, 2003

[7] J. Kong, J.H. Cui, D. Wu, and M. Gerla, "Building Underwater Ad-Hoc Networks and Sensor Networks for Large Scale Real-Time Aquatic Applications", *Proc of IEEE Military Communications Conference* (MILCOM), Oct 17-20, 2005, Atlantic City , New Jersey , USA.

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

[9] D. Pompili and T. Melodia, "Three-dimensional routing in underwater acoustic sensor networks", *Proc of the 2nd ACM Int'l Workshop on Performance Evaluation of Wireless Ad hoc, Sensor, and Ubiquitous Networks* (PE-WASUN2005), Oct 13, 2005, Montreal, Quebec, Canada.

[10] H. Pishro-Nik, N. Rahnavard, and F. Fekri, "Non-uniform error correction using low density parity-check codes", *IEEE Transactions on Information Theory*, Vol 51, No 7, Jul 2005.

[11] S. Mueller, R.P. Tsang, and D. Ghosal, "Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges", M.C. Calzarossa and E. Gelenbe (Eds): *MASCOTS 2003, LNCS 2965*, pp 209-234, ©Springer-Verlag Berlin Heidelberg 2004.

[12] Scalable Network Technologies, Inc., *Qualnet 3.8 Programmer's Guide*, 2005 [http://www.qualnet.com]

[13] A. Das, and D. Dutta, "Data Acquisition in Multiple-sink Sensor Networks", *Mobile Computing and Communications Review*, Vol. 9, No. 3, July 2005.

[14] E. Ilker Oyman and C. Ersoy, "Multiple Sink Network Design Problem in Large Scale Wireless Sensor Networks", *Proc of the IEEE International Conference on Communications* (ICC), Jun 20-24, 2004, Paris, France.