Multipath Virtual Sink Architecture for Underwater Sensor Networks

Winston K.G. Seah[1,2] **Hwee-Xian Tan**[2,1]
winston@i2r.a-star.edu.sg tanhweex@comp.nus.edu.sg

[1]Networking Department, Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613

[2]School of Computing, National University of Singapore
3 Science Drive 2, Singapore 117543

Abstract **– The salient features of acoustic communications render many schemes that have been designed for terrestrial sensor networks unusable underwater. We therefore propose a novel virtual sink architecture for underwater sensor networks that aims to achieve robustness and energy efficiency under harsh underwater channel conditions. To overcome the long propagation delay and adverse link conditions in such environments, we make use of multipath data delivery. While conventional multipath routing tends to lead to contention near the sink, we avoid this caveat with the virtual sink design involving a group of spatially diverse physical sinks. Hence, we are able to exploit the reliability achieved from redundancy provided by multipath data delivery while mitigating the contention between the nodes.**

I. INTRODUCTION

Terrestrial wireless sensor network technologies have progressed beyond research into actual deployment scenarios. Much of the sensor networking research has stressed on energy efficiency to maximize the network lifetime given the wireless communications characteristics and hardware constraints. Radio frequency (RF) or radio wave propagation suffers from severe attenuation in water and has been successfully deployed only at very low frequencies, involving large antenna and high transmission power. Hence, the current viable underwater physical layer technology is acoustic communications.

The salient features of acoustic communications render many schemes that have been designed for RF-based terrestrial sensor networks unusable. Besides having low bandwidth and a propagation delay five orders of magnitude higher than RF in air, the link quality also poses many challenges to underwater communications [1]. Underwater link quality is extremely volatile, and suffers frequent temporal disconnections due to numerous reasons, such as underwater current, temperature fluctuation, severe multipath fading, ambient noise and interference from marine life.

The high propagation delay makes it extremely inefficient to use automatic repeat request (ARQ) techniques that are commonly used in terrestrial networks for packet loss detection, and error recovery methods like retransmission incur excessive latency and signaling overheads. It would then appear that forward error correction (FEC) techniques can be applied to provide robustness against errors but at the cost of additional redundant bits competing for the already scarce bandwidth, and the processing needed for encoding and decoding further drains the critical energy resources.

We first present our *multipath virtual sink network architecture* and explain the design which is based on a novel approach of applying key conventional networking concepts. We validate the design by showing how simple protocols deployed in this architecture can significantly enhance the data delivery in a harsh underwater environment. Lastly, we conclude with discussion on ongoing research to design efficient protocols that are optimized for use in this architecture.

II. NETWORK ARCHITECTURE

The network topology is crucial in determining the network capacity, energy consumption, and more importantly, the reliability of the network. There must be sufficient robustness and redundancy built into the network to ensure that it continues to function even when a significant portion of the network is temporarily non-operational. It is envisaged that the underwater network topology will be made up of clusters of sensing nodes, where each cluster has one or more local aggregation points. These aggregation points will collectively form an underwater mesh network that connects to local sinks (as shown in Figure 1). Although we have shown a 2-tier topology, the number of tiers is flexible and can be dynamically adapted to meet deployment requirements and suit environmental conditions.

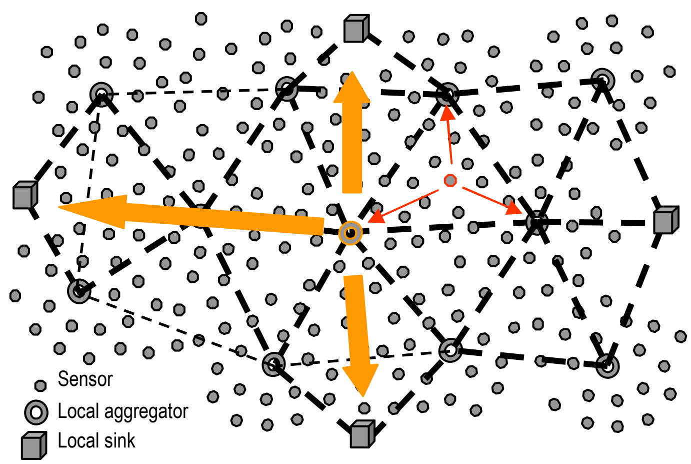

Figure 1: Proposed underwater network topology

A. *Virtual Sink*

It is assumed that the local sinks are connected via high-speed links (e.g. wired to a buoy on the surface equipped with RF communications link, or an undersea high-speed optical fibre) to a network where the resources are more than sufficient to support the communication needs of the various applications. Therefore, the ultimate goal of the underwater network is to ensure that data is delivered to one or more of these local sinks which collectively form a ***virtual sink***. As the sensing coverage is very dependent on the applications' needs and the technologies that are used to develop the sensors, we focus only on the communications aspects of sensors such as the range and bandwidth of the communication link.

B. *Multipath Data Delivery*

A robust multi-path data delivery scheme provides end-to-end connectivity to the local aggregation points. The scheme aims to maintain *n* routes to the neighbouring local aggregation points and provide local data caching; the value of *n* adapts to the channel conditions and also the criticality of the data carried in the packet. As the underwater channel is intermittent and bandwidth very limited, it may be better for the underwater nodes to cache data and transmit when the channel conditions are favourable rather than attempt multiple retransmissions. For time-critical data, instead of caching, the scheme will attempt to deliver data over more routes (larger *n* value) to increase the probability of successful delivery.

Similarly, the local aggregation points form a wireless mesh network that provides multiple paths to multiple local sinks which collectively form the virtual sink. Congestion at aggregation points (mesh nodes) can occur with simultaneous arrival of high traffic from sensor nodes, e.g. sensor data arising from the detection of the engine noise of a moving boat on the surface may generate a consecutive burst of sensor traffic arriving at neighboring aggregation points. As the name implies, in-network data aggregation is necessary to handle the congestion at the aggregation points. Likewise, the deployment of redundant nodes (as backup aggregation points) to increase the availability of multiple disjoint paths such that backup routes are readily available can be done, where necessary. This is crucial for sending time-critical delay-intolerant data that cannot be cached until the channel conditions improve. The multipath routing protocol will select the appropriate routes from those available to achieve the required service levels.

III. REDEFINING MULTIPATH AND RETRANSMISSION

Typical multipath routing protocols setup multiple routes between a pair of communicating nodes [2]. Depending on how the routes are selected, there is a strong likelihood of contention occurring among nodes that are on different routes but close to one another. As the routes converge at the destination node, the possibility of contention is even higher. Hence, the redundancy that multipath provides in the attempt to improve packet delivery is nullified by the contention among nodes, which can be made worse by retransmissions.

We therefore propose that a node (e.g. A in Figure 2) sends a packet simultaneously over spatially diverse routes to multiple sinks (S_1, S_2 and S_3), which form the virtual sink, and as long as a copy of the packet reaches one of these sinks, delivery is successful. This can be considered as "retransmitting" a packet simultaneously instead of sequentially, achieving lower latency and less packet transmissions, thus saving energy. The use of spatially diverse paths also reduces the possibility of contention.

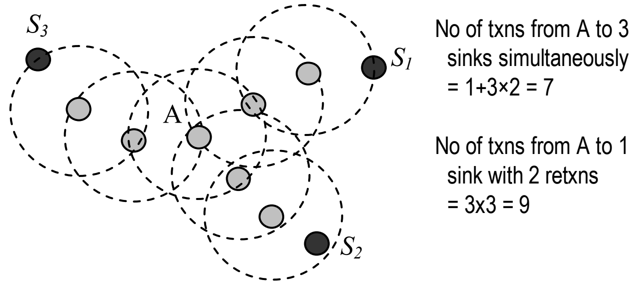

Figure 2: Multi-path Multi-sink/Virtual Sink

In the following subsections, we first describe a simple hopcount update mechanism that is performed during the network initialization phase for use in the data delivery mechanism, followed by the data delivery mechanism itself that we use together with our virtual sink architecture. These are just some of many possible data delivery schemes that can be applied in this architecture.

A. *Hopcount Update Mechanism*

During the setup/initialization phase, each sink broadcasts a hopcount update message to identify itself. When a sensor node receives this message, it will note the hopcount value (i.e., number of intermediate nodes that are used to forward the message) and rebroadcast the message after incrementing the value by one (hop). Every sensor keeps a record of its hopcount distances from all the sinks (initial values are set to a large number, e.g. 255. Consequently, a sensor that has no route to a particular sink will have 255 as the corresponding hopcount distance.) The propagation of hopcount information from sink S_0 is shown in Figure 3.

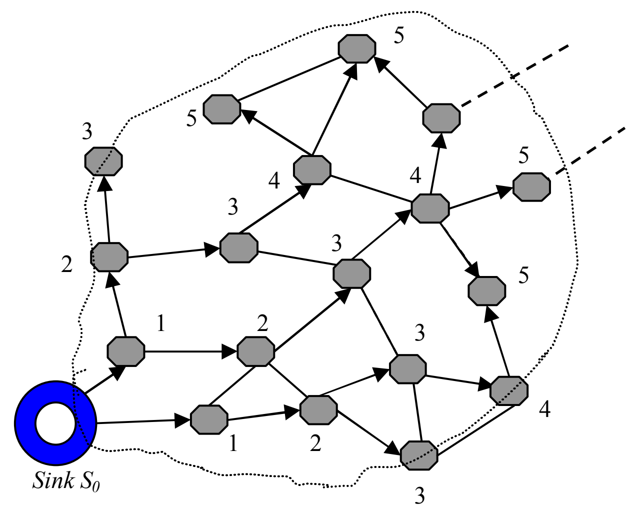

Figure 3: Propagation of hopcount information from S_0

B. *Data Delivery Mechanism*

The main caveat with existing routing mechanisms is that the wireless sensor network routing protocols are tailored for terrestrial networks, and cannot work well in underwater environments which have very different physical parameters. In addition, most of the current underwater routing protocols are centralized in nature, which is unsuitable for underwater ad hoc wireless networks that do not require centralized control. While AODV-BI [3] is a decentralized routing protocol that is designed for underwater ad hoc networks, it is unsuitable for underwater sensor networks because of its address-centric nature (data is sent from a source to a single destination). Underwater sensor networks are typically static in nature, subject to intermittent connectivity and are mainly involved in periodic oceanic monitoring applications (which are data-centric). AODV-BI will incur excessive overheads during route discovery and route maintenance. As such, there is a need to make use of a simpler and more efficient routing mechanism that is distributed, incurs little overhead and is data-centric.

In our proposed data delivery scheme, each node also keeps track of the node which it received the hopcount update from (also known as the "previous hop"), during the exchange of hopcount update messages. The previous hop provides information on the path back to each of the connected sinks in the network. Therefore, a node which has sensed data can deliver the packet to any one of the sink nodes that it is connected to, by sending it to the previous hop recursively. This is also known as reverse-path forwarding, a method which is similar to that being used in some terrestrial wireless sensor network routing protocols such as Directed Diffusion. As there are *n* sink nodes in the network, which are all connected via high speed optical fibers, the sensor nodes can forward the data to any of the *n* sinks that is connected to (e.g. S_0 as shown in Figure 4.)

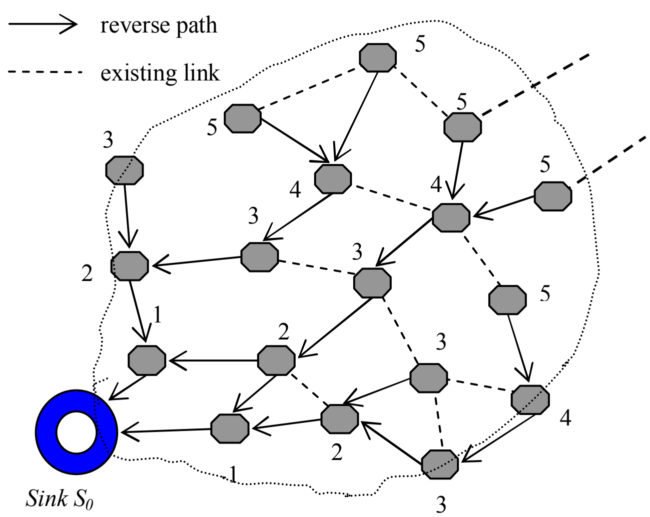

Figure 4: Reverse Path Forwarding to Sink S_0

IV. PERFORMANCE ANALYSIS

To validate the effectiveness and efficiency of our proposed multipath virtual sink architecture, we make use of simulations that are performed in Qualnet [4], which provides a scalable simulation platform for both wired and wireless networks. We consider an underwater region of 2.5km×2.5km to be monitored periodically for variations in physical parameters such as the temperature and salinity of the seawater. A total of 100 static sensor nodes are randomly deployed in the region to be monitored. The other simulation parameters are shown in Table 1.

Table 1: Simulation Parameters

Parameter	Value
Transmission range	400 meters
Channel frequency	15 kHz
Data rate	5000 bits/second
Speed of sound	1500 ms^{-1}
Propagation loss (spherical) [5]	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 static nodes send packets sequentially at time intervals of 100 seconds.

We use CBR as the traffic model to simulate the periodic monitoring of oceanic activities unverwater. All the 100 static nodes send data to the sinks periodically at time intervals of 100 seconds, with staggered initial starting times. For example, if Node 5 were to send a data packet at time t = 50.0s, Node 6 would send a data packet at time t = 51.0s. As such, the subsequent times that Nodes 5 and 6 send their next packets would be at t = 150.0s and t = 151.0s respectively. During each simulation run, the first 20 seconds are used as the initialization phase where hop count information from the sinks is propagated throughout the network to all the reachable static nodes in the network. Data traffic will commence only after the initialization phase, when no hop count information is sent.

A. Types of Forwarding Schemes

We simulate the following forwarding schemes and compare their performance in the simulator:

- 1 path with 1 retransmission – Data is sent along one path to a particular sink with hop-by-hop acknowledgements (ACKs). If the ACKs are not received within a specified time period, then the packet is retransmitted for a maximum of ***one*** time.
- 1 path with 2 retransmissions – This scheme is similar to the above scheme, except that data packets may be transmitted for a maximum of ***two*** times.
- 1 path with 3 retransmissions – As like the previous two schemes, data packets may be retransmitted if ACKs are not received after the waiting time. The maxmimum number of retransmissions is ***three***.
- 2 paths – Data packets are forwarded to the ***two*** nearest sinks from the source node, using ***two*** paths which may have overlapping intermediate hops. No ACKs or retransmissions are performed.
- 3 paths – Data packets are forwarded to the ***three*** nearest sinks from the source node, using ***three*** paths which may have overlapping intermediate hops. No ACKs or retransmissions are performed.
- 4 paths – Data packets are forwarded to the ***four*** nearest sinks from the source node, using ***four*** paths which may have overlapping intermediate hops. No ACKs or retransmissions are performed.

Each scheme is run for 20 times with different seeds, and the measurements are averaged to minimize any arbitrary randomness in the simulation.

B. *Sink Locations*

As described in the previous subsection, depending on the forwarding scheme being used, each data packet can be routed to one or more sinks. For the single-path scenarios, we assume that there is only one sink (Figure 5, Sink S_0) being placed at one corner of the network. For the multi-hop scenarios, we assume that there are a total of four sinks (Figure 5, Sinks S_0, S_1, S_2 and S_3) that are being placed in the network, with one sink at each corner – and that the network will dynamically select the shortest paths (or nearest sinks) to send data to when the traffic is generated. All the nodes in the network are connected to at least one sink via multiple hops; hence no partitions exist in the network during the simulations.

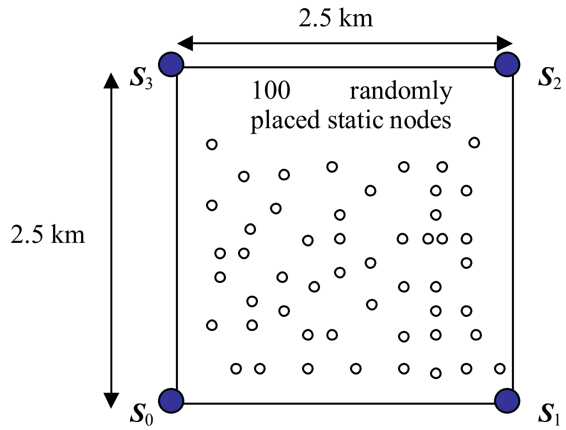

Figure 5: Sink Locations in the network

C. *Performance Metrics*

We study and evaluate the performance of the different forwarding schemes using the following set of metrics:

- Total number of packets forwarded – As the static nodes in the network may not be within the transmission range of the sink(s), data that is generated may have to travel through multiple hops before they can reach the sink(s). Hence, this metric measures the total number of packets that are forwarded by intermediate nodes before it reaches the sink.
- Total number of transmissions – This gives the total number of packet transmissions that take place throughout the network, for the simulated network lifetime. It includes any retransmissions, as well as data forwarding by the intermediate nodes.
- Total number of packets received by all the sinks – In the multipath schemes, different sinks may receive the same packet (identified using a unique sequence number) from the source nodes. Here, we do not differentiate between the data packets and merely compare the total number of data packets received by all the sinks in the network.
- Packet Delivery Ratio (PDR) – The PDR is given by the total number of ***unique*** packets received by all the sinks as a fraction of the total number of packets that are generated by the sinks. For example, if both sinks S_0 and S_1 receive data packet with sequence number 1 from the static source node 2, then the number of unique packets being received is considered as 1.
- Redundancy factor R_f – R_f is given by the total number of duplicate packets received as a fraction of the total number of unique packets received by all the sinks in the network. It gives a measure of the overall efficiency and redundancy of the schemes. A redundancy factor of 0 means that there there are no duplicate packets being received by the sinks. The higher the value of R_f, the higher the number of duplicated packets and the higher the amount of wasted resources in the network.
- Average end-to-end delay – The average end-to-end delay is the average ***shortest*** time taken for a packet to travel to any one of the sinks from the time of generation of the data packet.
- Total number of ACKs sent – This is the total number of one-hop ACKs that are being sent by the nodes in the network, upon the receipt of a data packet.
- Total number of retransmissions – Retransmissions may be performed by the source node or the intermediate forwarding node when it does not receive an ACK from the next forwarding node or expected destination. This can be due to either: (i) loss of the data packet, such that the ACK is never generated; or (ii) loss of the ACK packet.

D. *Results and Analysis*

We vary the per-hop Packet Loss Ratio (PLR) of the network from 0.05 to 0.50 and study the performance of the different schemes under difeerent PLR values. The comparative results are shown in Figure 6 to Figure 13.

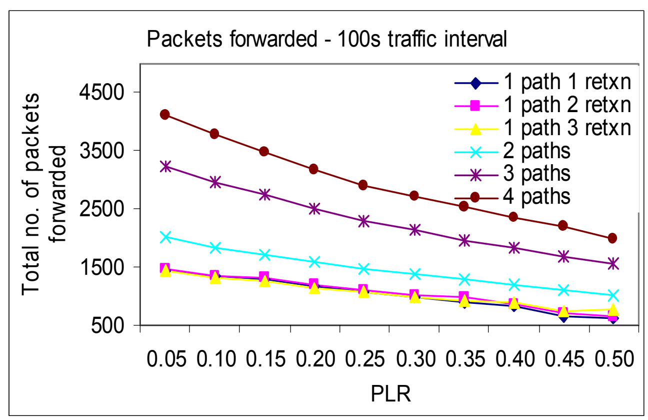

Figure 6: Packets forwarded vs PLR

Figure 6 shows the total number of packets that are being forwarded by the intermediate nodes before the data reaches the sink. A packet that is successfully sent over one hop is counted as one, even if it it needs to be retransmitted a few times. As the PLR increases, the total number of packets forwarded decreases because more packets are being dropped as they traverse through multiple hops. The total number of forwarded packets in the single path scenarios is less than that of the multiple path scenarios which sends multiple copies of the same data packet are simultaneously to multiple sinks. In the single path forwarding schemes, there is a pre-specified sink S_0, and the data packets from a particular node to the sink remains unchanged, as we are assuming a simple reverse path forwarding scheme.

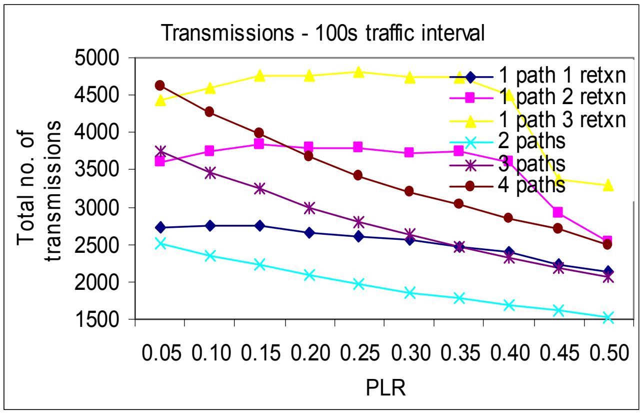

Figure 7: Number of transmissions vs PLR

The total number of transmissions in the network is shown in Figure 7. Under the multipath scenario, the total number of transmissions decrease with increasing PLR because the probability of dropped packets increases. In the single-path scheme, the number of transmissions decreases less gradually than that of the multipath scenarios because retransmission attempts are allowed for lost data packets. Hence, the approach of 1 path with 3 retransmission attempts has higher number of transmissions than that of 1 or 2 retransmissions because lost data packets are allowed to be retransmitted more times (resulting in higher number of transmissions; see also the results in Figure 13).

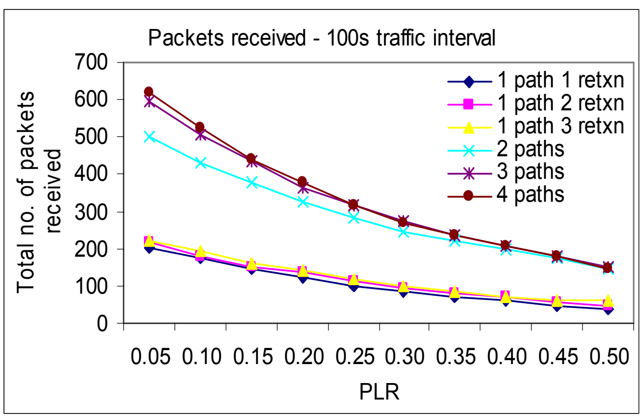

Figure 8: Packets received vs PLR

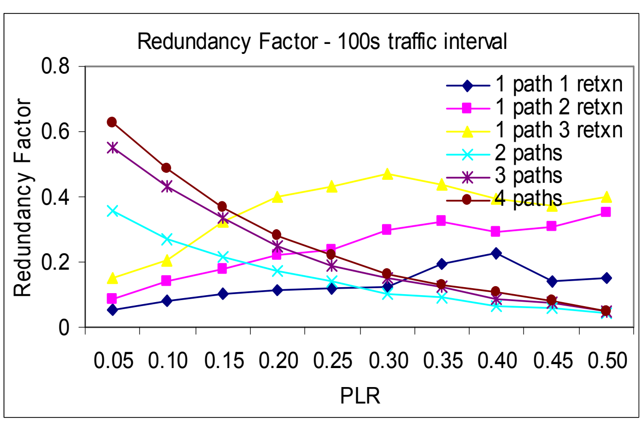

Figure 9: Redundancy Factor R_f vs PLR

Figure 8 and Figure 9 show the total number of packets being received by all the sinks and the redundancy factor $\boldsymbol{R}_f$, respectively. With more paths, it is expected that the sinks will receive more data packets and have higher $\boldsymbol{R}_f$ values due to the higher number of duplicate packets. When the PLR increases, the number of packets received and the $\boldsymbol{R}_f$ value decrease correspondingly because there is a higher probability of packets being dropped in the network. An interesting trend to note is that the $\boldsymbol{R}_f$ values for the multi-path scenarios appear to be lower than that of the single-path scenarios, although each data packet is simultaneously transmitted to more than one sink for the former case. An analysis of the trace files that were generated during the simulations revealed that this is due to the fact that many single-hop acknowledgemet packets (ACKs) are lost, even when the PLR is low. This is because the data packets traverse in one general direction, towards the sink $\boldsymbol{S_0}$, and these cause many collisions to occur when the ACKs are being sent. As the number of retransmission attempts is increased, the redundancy factor also increases because the loss of the ACKs results in more retransmissions and duplicated reception of data packets at the sink.

Figure 10 shows the PDR of the network, which decreases with increasing PLR, for all the different forwarding schemes. In general, the PDR is higher as $\boldsymbol{n}$ increases, where $\boldsymbol{n}$ ($1 \leq \boldsymbol{n} \leq 4$) is the number of paths that are used to forward the data packets to the sinks. The single-path scenarios have lower PDR than that of the multi-path scenarios due to the unnecesary retransmission of data packets resulting from lost ACK packets. When the number of permissible retransmissions is higher, there is a higher possibility of more redundant data duplicates in the network, resulting in more packet collisions and data loss.

The average end-to-end delay with respect to increasing PLR is shown in Figure 11. In single-path forwarding schemes, increased PLR results in increased delay, because of the retransmission of data packets during packet losses. The higher the number of permissible retransmissions, the higher the delay, because each retransmission has to wait for a certain amount of delay (to ascertain that the packet has been lost). Under the multi-path schemes, the delay appears to be quite constant and independent of the PLR, because the source nodes sends multiple copies of a packet to different sinks, and only the shortest end-to-end delay (of the first copy to reach a sink(is being considered during the delivery.

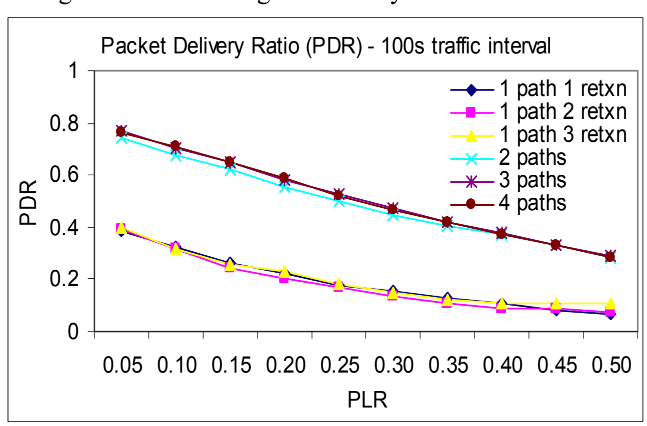

Figure 10: Packet Delivery Ratio (PDR) vs PLR

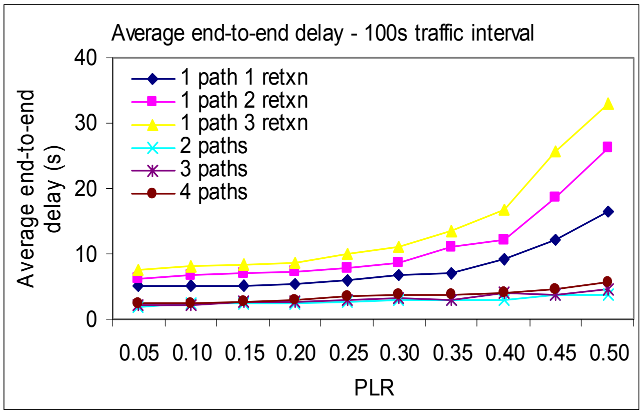

Figure 11: Average end-to-end delay vs PLR

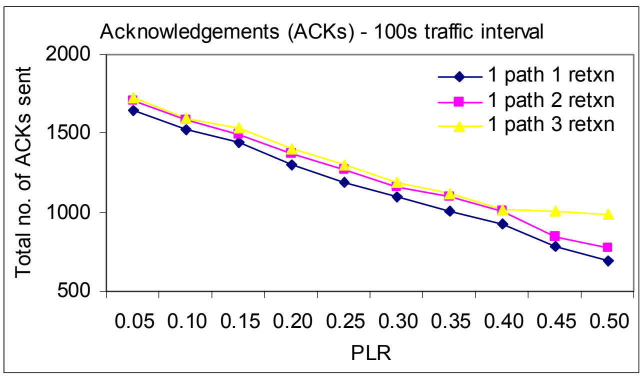

Figure 12: Number of acknowledgements (ACKs) vs PLR

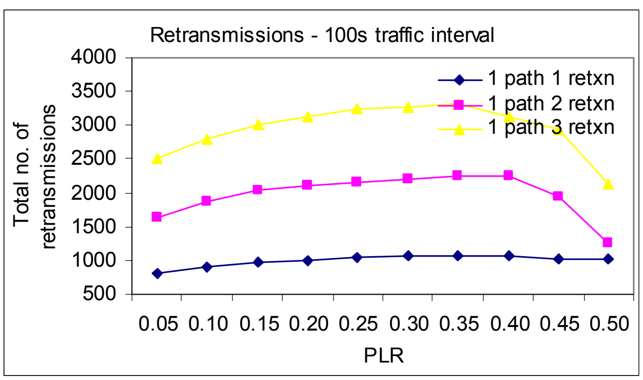

Figure 13: Number of retransmissions vs PLR

Figure 12 and Figure 13 show the number of ACKs and retransmissions that are being generated in the network, respectively. As the PLR increases, the number of ACKs decreases because there is a higher number of lost data packets. However, the number of retransmissions remains quite constant despite the change in PLR values, because it is also more likely for ACKs to be lost although the corresponding data packets have been received successfully by the intended destination or intermediate forwarding nodes (as compared to data packets, since they are traversing in opposite directions) even during low PLR values. Therefore, during low PLR, the retransmission attempts are caused by unnecessary retransmissions arising from lost ACKs, while during high PLR, the retransmissions are caused by lost data packets. In general, the total number of ACKs and retransmissions are higher for the single-path forwarding schemes with with higher number of permissible retransmissions.

V. CONCLUSION AND FUTURE WORK

As increasingly more applications (such as environmental monitoring and surveillance activities) are being deployed in the harsh underwater environment, it becomes a necessity to re-evaluate the applicability and performance of the conventional single-path single-sink architecture that is being deployed in most terrestrial wireless sensor networks. The main problem with the traditional architecture is that bottlenecks can quickly form at the regions around the sink, and the sensor nodes around the sink are usually also susceptible to node failures because they are being used to transmit data to the sinks more frequently.

As such, we have proposed the use of a virtual sink architecture, in which sensor nodes can forward data to one or more spatially diverse sinks to avoid contention and achieve high reliability despite the adverse network conditions. In this paper, we have used extensive simulations to compare the performance of our approach against that of the single-path single-sink with hop-by-hop ACKs and a finite number of permissible retransmissions. From our results, we have shown that it is indeed possible to achieve better network performance (in terms of PDR and average end-to-end delays) with slightly more overheads (in terms of the redundancy ratio) using our proposed virtual sink architecture.

As part of our future work, we will be looking into the optimum number of ***n***-paths that can be used in varying network conditions, as well as the feasibility of dynamically adjusting the ***n***-value to suit the prevailing channel conditions. This aims to reduce the number of redundant transmissions, while achieving robustness and reliable data delivery.

REFERENCES

[1] M. Sozer, M. Stojanovic and J. G. Proakis, "Underwater Acoustic Networks", *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 72-83, Jan 2000.

[2] 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, 2004. ©Springer-Verlag Berlin Heidelberg 2004.

[3] K. Y. Foo, P. R. Atkins, T. Collins, C. Morley and J. Davies, "A Routing and Channel-Access Approach for an Ad Hoc Underwater Acoustic Network", Proceedings of MTS/IEEE OCEANS '04, Kobe, Japan, 9 – 12 Nov, 2004.

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

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