Universidad de Málaga

Escuela Técnica Superior de Ingeniería de Telecomunicación

TESIS DOCTORAL

Self-organizing Fast Routing Protocols for Underwater Acoustic Communications Networks

Autor:

Waheeduddin Hyder

Directores:

Pablo Otero Roth

Miguel Ángel Luque Nieto

Málaga 2020

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

D./Dña WAHEED UDDIN HYDER
Estudiante del programa de doctorado INGENIERÍA DE TELECOMUNICACIONES de la Universidad de Málaga, autor/a de la tesis, presentada para la obtención del título de octor por la Universidad de Málaga, titulada: SELF-ORGANIZING
FAST ROUTING PROTOCOLS FOR UNDERWATER ACOUSTIC COMMUNICATIONS NETWORKS

Realizada bajo la tutorización de PABLO OTERO ROTH y dirección de PABLO OTERO ROTH, MIGUEL ÁNGEL LUQUE NIETO (si tuviera varios directores deberá hacer constar el nom re de todos)

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 5 de MARZO de 2020

Fdo.: WAHEED UDDIN HYDER

Acknowledgement

I would like to express my heartfelt gratitude to The Late Dr. Javier Poncela without whom I could have never achieved my goal. He helped me remarkebly throughout the research work and made it look much easier than I thought. He was intrumental in defining the path of my research and guided me on every stage with his constructive critiques and suggestions.

I am also very thankful to Prof. Dr. Pablo Otero for his valued guidance and critical assessment of the reaserch work. Having vast knowledge and research experience, he guided me to work on very fine aspects of the research, which made it quite valuable.

Last but not the least, I am very grateful to Dr. Miguel Ángel Luque for his help and guidance. After the sad demise of Dr. Javier Poncela, Dr. M.A. Luque and Dr. P. Otero helped me tremandously. With their sincere help and effort I was able to continue my reasearch work without any disruption.

I am thankful to my family and friends for their support. I could not have achieved this major goal without my parents prayers, my wife Amna's tremendous support and my friends Bilal, Sabeen, Nayyer, Anwer and Zeeshan's help.

Abstract

Underwater Wireless Sensor Networks (UWSNs) constitute an emerging technology for marine surveillance, natural disaster alert and environmental monitoring. For terrestrial wireless networks we use electromagnetic (EM) waves for communication over the air. Unlike terrestrial Wireless Sensor Networks (WSNs), EM waves cannot propagate more than few meters in water due to the high absorption rate. However, acoustic waves can travel long distances in underwater compared to EM waves. Therefore, instead of EM waves, acoustic waves are preferred for underwater communications. Acoustic waves are more suitable for underwater communication, but they travel very slow compare to EM waves. Typical speed of acoustic waves in water is 1500 m/s whereas speed of the EM waves in air is approximately $3x10^8$ m/s. Therefore, the terrestrial WSN protocols assume that the propagation delay is negligible and the reactive routing protocols are deemed acceptable for WSNs. The propagation delay which is caused by the slow propagation speed of acoustic waves in water is a natural phenomenon and cannot be reduced but we can do otherwise and try to reduce the end-to-end packet delays. Routing delay is one of the major factors in end-to-end packet delay. In reactive routing protocols, when a packet arrives to a node, the node takes some time to select the node to which the data packet would be forwarded. This delay becomes substantial for time-critical applications. We may reduce the routing delay for time-critical applications by using proactive routing protocols.

Other critical issues related to UWSNs are determining the position of the underwater nodes and time synchronization among the nodes. Wireless sensor nodes need to determine the position of the surrounding nodes to select the node that will forward the packet towards the sink node. Underwater nodes cannot determine their position using a Global Navigation Satellite System (GNSS) because of the very short underwater range of the GNSS signal. Timestamping to determine the distance between two nodes may be used in WSNs but the limited mobility of the UWSN nodes and variation in the propagation speed of the acoustic waves make the time synchronization a challenging task for underwater acoustic networks. For all these reasons, WSN protocols cannot be readily used for underwater acoustic networks.

To address this problem, routing strategies focused on different aspects have been proposed: location free, location based, opportunistic, cluster based, energy efficient, etc. These mechanisms usually require measuring additional parameters, such as the angle of arrival of the signal or the depth of the node, which makes them less efficient in terms of energy conservation. In this thesis, we propose a cross-layer proactive routing initialization mechanism that does not require additional measurements and, at the same time, is energy efficient. We have proposed two routing protocols

for different topologies. Self-Organized Fast Routing Protocol for Radial Underwater Networks (SOFRP) is for radial topology and Self-organized Proactive Routing Protocol for Non-uniformly Deployed Underwater Networks (SPRINT) is for a network in which the nodes are unevenly and randomly deployed.

SOFRP is based on the algorithm to recreate a radial topology with a gateway node, such that packets always use the shortest possible path from source to sink, thus minimizing consumed energy. Collisions are avoided as much as possible during the path initialization. The algorithm is suitable for 2D or 3D areas, and automatically adapts to a varying number of nodes, allowing one to expand or decrease the networked volume easily.

In SPRINT the routing path from the node to the gateway is formed on the basis of the distance. The data sending node prefers to choose the neighbor node which is closest to it. The distance is measured by the signal strength between the two nodes. It is designed to achieve high data throughput and low energy consumption of the nodes. There is a tradeoff between the throughput and the energy consumption in the wireless networks. The transmission energy depends on the distance between the communicating nodes. To successfully send a packet more energy is consumed for longer distance. On the other hand, the number of relay nodes or hops between the source node and the destination node is a key factor which affects the throughput. Each hop increases the delay in the packet forwarding and as a result decreases the throughput. Hence, energy consumption requires nearest nodes to be chosen as forwarding node whereas the throughput requires farthest node to be selected to minimize the number of hops.

Resumen

Las redes subacuáticas de sensores (UWSN) constituyen una tecnología emergente para aplicaciones como la vigilancia submarina, la predicción de desastres naturales o la monitorización medioambiental. En redes terrestres de sensores (WSN) se utilizan ondas electromagnéticas (EM) como portadoras de las comunicaciones, Sin embargo, la propagación de ondas EM en el agua sufre una intensa atenuación por lo que las distancias alcanzadas no son prácticas. Las ondas acústicas, en cambio, se propagan con menos atenuación por lo que sus alcances prácticos son mucho mayores y por ello son las que se usan generalmente en UWSNs. La velocidad de propagación acústica típica en agua es de 1500 m/s, es decir, unas 200000 veces menor que la de la onda EM en el aire. En el diseño de los protocolos comúnmente utilizados en WSNs es correcto asumir que los retardos de propagación son mucho menores que otras demoras en la transmisión, lo que hace que los protocolos reactivos de encaminamiento se consideren adecuados. El retardo de propagación en comunicaciones acústicas es un fenómeno natural que no puede evitarse y, en muchas ocasiones, mayores que otros retardos de transmisión, por lo que para reducir el retardo extremo-a-extremo sólo cabe actuar sobre los protocolos de encaminamiento, que es una de sus principales causas. En los protocolos reactivos de encaminamiento, cuando un paquete de datos es recibido por un nodo de la red, el nodo necesita de cierta cantidad de tiempo para elegir el nodo al que se re-enviará el paquete. Ese tiempo significa un retardo que aumenta el retardo extremo-a-extremo, que puede llegar a exceder el máximo permitido en algunas aplicaciones. Una posible estrategia para reducir ese retardo en aplicaciones donde es crítico consiste en utilizar protocolos proactivos de encaminamiento.

Otros asuntos críticos relacionados con UWSNs son la determinación de las posiciones de los nodos subacuáticos y la sincronización temporal de los relojes de los nodos. Conocer la posición del resto de los nodos es imprescindible para decidir a qué nodo se re-envía un paquete. En redes terrestres los nodos conocen su posición y se sincronizan con ayuda de receptores GNSS (Global Navigation Satellite System). En redes subacuáticas no hay señal GNSS debido a la muy escasa penetración en el agua de las ondas EM de las frecuencias utilizadas por esos sistemas. En cuanto a la determinación de las distancias entre nodos, si bien en redes terrestres es posible utilizar marcas temporales si existe sincronización entre nodos, en redes subacuáticas la velocidad de propagación no es constante, lo que complica la determinación de distancias entre nodos y la sincronización. Todas estas realidades hacen que no puedan utilizarse directamente los protocolos comunes en WSNs en redes subacuáticas.

Para superar estas dificultades en la literatura técnica se han presentado trabajos que recurren a distintas estrategias (que se explican en el Capítulo 2 de esta memoria) que hacen uso de informaciones obtenidas en los nodos de ciertos parámetros, como son el ángulo de llegada a un nodo de la onda acústica portadora o la profundidad a la que se encuentra el nodo. La necesidad de utilizar sensores adicionales para la medida de estas magnitudes disminuye la eficiencia energética de los nodos, debido al consumo de esos sensores. En esta tesis se propone un mecanismo *cross-layer* de inicialización de encaminamiento proactivo que no necesita de medidas adicionales y que, al mismo tiempo, es eficiente en términos de energía. En este trabajo se presentan dos protocolos de encaminamiento para distintas topologías de la red. El que hemos denominado *Self-Organized Fast Routing Protocol for Radial Underwater Networks* (SOFRP) se utiliza para redes de topología radial y el denominado *Self-organized Proactive Routing Protocol for Non-uniformly Deployed Underwater Networks* (SPRINT) se ha pensado para redes en los que los nodos están distribuidos aleatoriamente en un determinado volumen subacuático.

SOFRP está basado en un algoritmo que recrea una topología radial alrededor de un nodo de enlace ("Gateway", en adelante), de manera que los paquetes siempre usan el camino más corto entre nodo fuente del paquete y el nodo Gateway, de manera que se minimiza la energía consumida en la transmisión de los paquetes. Las previsibles colisiones entre paquetes debidas al uso del medio compartido se evitan en la medida de lo posible en el proceso de inicialización. El algoritmo es adecuado para regiones bi/tri-dimensionales (2D o 3D) y se adapta automáticamente a un número variable de nodos, lo que permite el cambio de la dimensión de la red.

En SPRINT, la ruta entre el nodo fuente del paquete y el Gateway se construye sobre la base de la menor distancia entre el nodo relevador y todos sus nodos vecinos. El nodo transmisor elige de entre todos sus nodos vecinos aquel que se encuentra más cerca de él. Las distancias se estiman a partir de las potencias recibidas en los nodos. El protocolo se ha diseñado para conseguir grandes cantidades de datos transmitidos ("Throughput", en adelante) con pequeños consumos de energía en los nodos. Un objetivo de diseño es el compromiso de equilibrio entre el Throughput y el consumo energético, porque la energía necesaria para la transmisión aumenta con el alcance necesario y, por consiguiente, con la distancia del enlace entre dos nodos. Pero, por otro lado, la elección de nodos más cercanos hará que sean necesarios más saltos o relevos para que un paquete progrese de su nodo fuente al Gateway y esto provoca la disminución del Throughput porque aumenta el retardo extremo-a-extremo. Es decir, la eficiencia energética prefiere la elección de nodos cercanos mientras que el Throughput prefiere menos relevos. El protocolo presentado busca encontrar el equilibrio entre ambos criterios.

Table of Contents

CHAPTER 1 INTRODUCTION 1

1.1 RESEARCH BACKGROUND 1

1.2 INTRODUCTION TO TWSN 1

1.3 TWSN PERFORMANCE REQUIREMENTS 2

1.4 INTRODUCTION TO UWSN 3

1.5 UWSN APPLICATIONS 5

1.5.1 Military Applications 5

1.5.2 Disaster Monitoring 7

1.5.3 Oceans and Rivers Environmental Monitoring 7

1.6 BEHAVIOR OF UNDERWATER ACOUSTIC WAVES 10

1.6.1 Sound Velocity 10

1.6.2 Multipath Fading 12

1.6.3 Transmission Loss 13

1.6.4 Channel Noise 16

1.6.5 Relationship between the Bandwidth Capacity and the Transmission Distance 16

1.7 UWSN ARCHITECTURE 17

1.7.1 Two-Dimensional Architecture 17

1.7.2 Three-Dimensional Architecture 18

1.8 UWSN ISSUES 20

1.8.1 Time Synchronization and Localization 20

1.8.2 Deployment and Replacement of Sensor Nodes 21

1.9 THESIS OBJECTIVES 21

1.10 CONCLUSION 22

1.11 THESIS ORGANIZATION 22

CHAPTER 2 STATE OF THE ART 25

2.1 INTRODUCTION 25

2.2 LOCATION FREE PROTOCOLS 25

2.3 LOCATION BASED PROTOCOLS 26

2.4 OPPORTUNISTIC PROTOCOLS... 28

2.5 CLUSTER BASED PROTOCOLS 30

2.6 ENERGY EFFICIENT PROTOCOLS.. 31

2.7 RELIABLE DATA DELIVER PROTOCOLS 32

2.8 PROTOCOLS BASED ON RECEIVED SIGNAL STRENGTH 33

2.9 CONCLUSION.. 34

CHAPTER 3 SOFRP: SELF ORGANIZED FAST ROUTING PROTOCOL 37

3.1 INTRODUCTION 37

3.2 NETWORK TOPOLOGY.. 38

3.3 SOFRP OVERVIEW 40

3.4 ROUTING PATH FORMATION 42

3.4.1 Phase 1: Search for Gateway Neighbors 42

3.4.2 Phase 2: String Formation 44

3.5 ADAPTATION FOR HEXAGONAL AND GRID TOPOLOGIES 49

3.6 STRING FORMATION IN CASE OF NODE FAILURE.. 51

3.6.1 Scenario 1: A Node is down at the Time of Routing Initialization............................. 51

3.6.2 Scenario 2: What will Happen If a Node in a String Goes down after the Initialization? 54

3.7 ROUTING PATH PERIODIC UPDATE PROCESS 55

3.8 PACKET FORMAT.. 56

3.9 PERFORMANCE ANALYSIS 57

3.10 COMPUTER SIMULATION RESULTS 61

FIGURE 3-23. ENERGY CONSUMPTION VS. NUMBER OF TIME SLOTS. 66

3.11 DISCUSSION OF RESULTS 67

3.12 CONCLUSION... 68

CHAPTER 4 SPRINT: SELF-ORGANIZED PROACTIVE ROUTING PROTOCOL FOR NON-UNIFORMLY DEPLOYED UNDERWATER NETWORKS **69**

4.1 INTRODUCTION 69

4.2 OVERVIEW OF THE SPRINT PROTOCOL 71

4.3 ROUTE FORMATION PROCESS 76

4.3.1 Forwarding the RR Packet 76

4.4 SELECTION CRITERIA 77

4.5 MATHEMATICAL ANALYSIS 79

4.6 PACKET HEADER FORMAT 80

4.7 COMPUTER SIMULATION RESULTS 80

4.8 CONCLUSIONS 87

CHAPTER 5 CONCLUSIONS AND FUTURE WORK **88**

5.1 OBJECTIVES REACHED IN THIS THESIS 91

5.2 FUTURE WORK 91

SUMMARY OF AUTHOR'S CV **100**

Figures

FIGURE 1-1. A TYPICAL DRILLING RISER [4] 4

FIGURE 1-2. 2H DATA LOGGER [5] 4

FIGURE 1-3. OEX AUVS [14] 6

FIGURE 1-4. EFOLAGA AUVS EQUIPPED WITH KM MODEMS [15] 6

FIGURE 1-5. GITEWS NETWORK [16] 7

FIGURE 1-6. DART NETWORK [17] 8

FIGURE 1-7. SENSOR NODE [21] FIGURE 1-8. GATEWAY [21] ..... **¡ERROR! MARCADOR NO DEFINIDO.**

FIGURE 1-9. MOTION CHARACTERISTICS OF USVS, AUVS, AND AUGS [22] 10

FIGURE 1-10. MULTIPATH FADING IN UNDERWATER ACOUSTIC COMMUNICATION [26] 13

FIGURE 1-11. ABSORPTION COEFFICIENT A(F) VS. FREQUENCY [29] 15

FIGURE 1-12 SPHERICAL SPREADING: THE ACOUSTIC INTENSITY DECREASES WITH DISTANCE FROM THE SOURCE. [30] 15

FIGURE 1-13. FREQUENCY VS SNR [29] 17

FIGURE 1-14. TWO DIMENSIONAL UWSN ARCHITECTURE [31] 18

FIGURE 1-15. THREE DIMENSIONAL ANCHORED NODE [31] 19

FIGURE 1-16. THREE DIMENSIONAL BUOY SUPPORTED SENSOR NODES [32] 19

FIGURE 1-17. AUV BASED UWSN NETWORK [33] 20

FIGURE 3-1. A 3D SPHERICAL WIRELESS SENSOR NETWORK. 37

FIGURE 3-2. NETWORK TOPOLOGY OF N STRINGS 39

FIGURE 3-3. EXAMPLE OF NETWORK WITH THREE STRINGS 40

FIGURE 3-4. TRANSMISSION SEQUENCE OF SGN, SHN, SHN_ACK, SGN_RSP AND SGNRSP_ACK BETWEEN THE GATEWAY AND THE NEIGHBORS 44

FIGURE 3-5. THE SAME TRANSMISSION SEQUENCE IN MESSAGE SEQUENCE DIAGRAM. 44

FIGURE 3-6. TRANSMISSION SEQUENCE OF RR, RR_ACK BETWEEN THE GATEWAY AND THE NEIGHBORS AND TRANSMISSION OF RR, RR_ACK, CF_RR, CFRR_ACK, CFRR_BACK AND CFRRBACK_ACK BETWEEN THE LAYERS AND THE ADJACENT NODES. 48

FIGURE 3-7. THE SAME TRANSMISSION SEQUENCE IS ALSO SHOWN IN MESSAGE SEQUENCE DIAGRAM. 48

FIGURE 3-8. PROPAGATION OF THE RR_RSP MESSAGE. 49

FIGURE 3-9. HEXAGONAL TOPOLOGY. 50
FIGURE 3-10. 2D GRID. 50
FIGURE 3-11. 3D GRID. 51
FIGURE 3-12. A NODE IS DOWN AT THE TIME OF ROUTING INITIALIZATION. 52
FIGURE 3-13. SEQUENCE OF MESSAGES TO DETERMINE THE CORRECT NODE FOR STRING FORMATION. 53
FIGURE 3-14. ALTERNATIVE PATH WHEN A NODE FAILS AFTER THE ROUTING PATHS HAVE BEEN FORMED. 54
FIGURE 3-15. PACKET HEADER FORMAT 56
FIGURE 3-16. CONVERGENCE DELAY VS. NO. OF STRINGS. 60
FIGURE 3-17. CONVERGENCE DELAY VS. NO. OF NODES. 60
FIGURE 3-18. NUMBER OF RETRANSMITTED PACKETS VS. PROBABILITY OF PACKET LOSS. 62
FIGURE 3-19. PERCENTAGE OF COLLISION RATE VS. NO. OF STRINGS. 63
FIGURE 3-20. COLLISION RATE VS. NUMBER OF NODES. 64
FIGURE 3-21. COLLISION RATE VS. CONVERGENCE DELAY. 65
FIGURE 3-22. COMPARISON OF TIME SLOTS FOR COLLISION RATE AND CONVERGENCE DELAY. 66
FIGURE 3-23. ENERGY CONSUMPTION VS. NUMBER OF TIME SLOTS. 66
FIGURE 4-1. 3D UWSN ARCHITECTURE 70
FIGURE 4-2. SELECTION OF RELAY NODES 70
FIGURE 4-3. ABSORPTION COEFFICIENT VS FREQUENCY 72
FIGURE 4-4. SNR VS FREQUENCY AND DISTANCE 72
FIGURE 4-5. NON-UNIFORMLY DEPLOYED NODES IN 2D NETWORK 74
FIGURE 4-6. EXAMPLE OF THREE METRICS-DISTANCE, NEIGHBORS, AND HOPS. 75
FIGURE 4-7. A TYPICAL COLLISION DOMAIN. 76
FIGURE 4-8. RR TRANSMISSION RANGE. 78
FIGURE 4-9. RR PACKET SENT BY NODE A. 78
FIGURE 4-10. RR PACKET SENT BY NODE B. 78
FIGURE 4-11. PACKET HEADER FORMAT 80
FIGURE 4-12. NUMBER OF HOPS AGAINST RSS WEIGHTS. 81
FIGURE 4-13. NUMBER OF HOPS AGAINST HOPS WEIGHTS. 82
FIGURE 4-14. NUMBER OF HOPS AGAINST NUMBER OF NEIGHBORS WEIGHTS. 82

FIGURE 4-15. DELAY AGAINST RSS WEIGHTS. 83
FIGURE 4-16. DELAY AGAINST HOPS WEIGHTS. 83
FIGURE 4-17. DELAY AGAINST NUMBER OF NEIGHBOR NODES 83
FIGURE 4-18. ENERGY CONSUMPTION PER NODE PER PACKET FOR 100 NODES. 84
FIGURE 4-19. ENERGY CONSUMPTION PER NODE PER PACKET FOR 200 NODES. 84
FIGURE 4-20. ENERGY CONSUMPTION PER NODE PER PACKET FOR 300 NODES. 85
FIGURE 4-21. ENERGY CONSUMPTION PER NODE PER PACKET FOR 400 NODES. 85
FIGURE 4-22. ENERGY CONSUMPTION PER NODE PER PACKET FOR 500 NODES. 85
FIGURE 4-23. ENERGY CONSUMPTION PER NODE PER PACKET FOR 600 NODES. 85
FIGURE 4-24. AVERAGE ENERGY CONSUMPTION. 86
FIGURE 4-25. RECRP ENERGY CONSUMPTION. 86

Tables

TABLE 3.1. PACKET DESCRIPTIONS 56
TABLE 3.2. LIST OF EQUATION SYMBOLS 57
TABLE 3.3. SIMULATION PARAMETERS 61
TABLE 3.4. PACKET DELAY ANALYSIS 62
TABLE 3.5. CONVERGENCE DELAY ANALYSIS 62
TABLE 4.1 PACKET DESCRIPTIONS. 73
TABLE 4.2. LIST OF THE SYMBOLS USED IN THE EQUATIONS 73
TABLE 4.3. SIMULATION PARAMETERS 80
TABLE 4.4. WEIGHTS COMBINATIONS USED IN THE SIMULATION. 81
TABLE 4.5. TRANSMISSION RANGES FOR DIFFERENT NUMBER OF NODES. 84
TABLE 4.6. SPRINT AND RECRP ENERGY CONSUMPTION COMPARISON. 87

Chapter 1
Introduction

1.1 Research Background

The idea of monitoring the environment or area surveillance is in practice for many decades. The electronic sensors, connected through wires, are commonly used for these purposes. In 1990s the idea of wireless sensor networks was evolved. The history of Terrestrial Wireless Sensor Networks (TWSNs) can be traced back to Distributed Sensor Networks (DSNs) program of Defense Advanced Research Projects Agency (DARPA) around 1980 [1]. In 1999 a wireless network of small sensors was proposed by DARPA's research project known as Smart Dust [2] . The main objective of Smart Dust was to design a sensor which is very small in size and consumes very low power to make it inexpensive and easy-to-deploy. Since then TWSN has been developed for many applications such as Mobile Adhoc Networks (MANETs), Body Area Networks (BANs) and Underwater Wireless Sensor Networks (UWSNs).

There is a significant difference between the TWSN and UWSN. TWSN networks use electromagnetic (EM) waves whereas UWSNs commonly use acoustic waves. Electromagnetic waves are not suitable for underwater communication because of high absorption rate in water. For underwater communication, acoustic waves are more suitable because of long propagation distance. However, underwater, the propagation speed of sound wave is around 1500 m/s, which causes very high propagation delay. The protocols for TWSN are designed with the assumption that the propagation delay is negligible. However, the propagation delay is significant in underwater networks which may increase collisions between the packets and decrease the throughput. Therefore, the challenges involved in UWSN are very much different from TWSN networks and the protocols designed for TWSNs cannot be used for UWSNs. Let's have a brief overview of TWSN to understand the pertinent goals and issues.

In addition to high propagation delay, acoustic waves have very low data rate as well, because practical frequency bands in underwater acoustic propagation are what in radiofrequency (RF) are called Very Low Frequency (VLF: 3-30 kHz) and Low Frequency (LF: 30-300 kHz).

1.2 Introduction to TWSN

TWSNs have been deployed for commercial and military purposes for about a decade now. In late 1990s Smart Dust project at University of California at Berkley developed small sensor nodes called motes. The goal of the project was to develop sensor devices of very small size like a grain of sand or a dust particle. It was also required that the sensor nodes should be extremely low-cost to be deployed in harsh environment in abundance and should not be replaced once their operational life is over. The required operational life depends on the type of application. For example, monitoring of the climate change may require years of operational life whereas monitoring of a crop may continue for months only. That requires the sensor nodes to consume the battery power as low as possible for a long operational life. Moreover, TWSNs were required to be self-configured and self-organized. TWSN has a wide variety of civilian and military applications. Some of the applications are given below.

1. Building Control Systems: These include, but not limited to, automatic meter reading, smoke detection, lighting control and HVAC controls. Sensing these systems allows their flexible management.
2. Industrial Automation: To detect critical failure and avoid the large-scale damage due to that failure.
3. Military Surveillance: To detect an unauthorized intrusion in a large open area.
4. Environmental Monitoring: TWSN can be used for monitoring of sensitive wildlife and habitats nonintrusively and nondestructively.
5. Structural Monitoring: To detect the structural hazards and conduct the nondestructive tests continuously for buildings and bridges.

When TWSN was proposed, the existing medium access control (MAC) and routing protocols, either for wired network or wireless network, were not designed with the goal to consume low power. This new requirement of TWSNs leads to design new medium access and routing protocols to provide high throughput and low latency along with the low power consumption.

1.3 TWSN Performance Requirements

1. Energy Efficiency: As stated earlier, the TWSN nodes are usually deployed in abundance and unattended environments which makes the battery replacement difficult. Although in some environments the solar power or other alternative power sources may be used to energize the nodes or charge the batteries to extend their lifetime, the overall lifetime of the sensor nodes is still limited. Due to these constraints, the low power consumption is the prime concern for TWSNs. At physical layer, the power consumption can be reduced by employing low-power transmission signal, less complex channel coding scheme and low-power electronics. At MAC layer, the power consumption can be reduced by avoiding packets retransmission (in case of collision or error), large control packet overhead, idle listening, and overhearing.

2. Throughput: The high throughput is usually the most desired parameter for any data network. Header size of the packet, the number of control packets, type of duplex transmission and the channel bandwidth are the factors which affect the throughput.
3. Latency: Latency or delay to forward a data packet from the sender node to the destination node becomes important for mission critical networks, like fire alarm system for buildings to activate automatic shower in case of fire accident or pressure sensor of a boiler to activate the safety valve immediately in case of the pressure goes above the threshold value. In cooperative networks, the intermediate nodes which forward the data packet to the destination node increase the delay in the packet delivery. In addition to that, the packet length affects the delay as it takes longer to transmit a large size packet than a small size packet. Retransmissions in case of collisions in contention-based MAC protocols also incur delay.
4. Robustness: The robustness describes a network's behavior in case of a node failure or connectivity failure. The node failure is called the node percolation and the connectivity failure is called the edge percolation. The degree of a node is the number of connections it has with the neighbor nodes. By counting how many nodes have each degree, we get the degree distribution. The degree distribution between the nodes is an important factor for the robustness of a network.
5. Stability: A networks ability to handle the traffic load fluctuations over a sustained period of time is known as stability.
6. Scalability: Scalability refers to the networks ability to maintain its functions according to its performance parameters when the size of the network increases.
7. Fairness: Sharing of the resources among the nodes should not be biased.

1.4 Introduction to UWSN

The traditional method for underwater monitoring was based on manual deployment of sensor nodes, which were equipped with the data recorder. The data was gathered by recovering the sensors nodes after sometime [3]. This way, real-time monitoring was not possible. In addition to the delay in data gathering, the node failures that may occur during the deployment of the nodes could not have been detected promptly. It was also possible that the recoded data was lost during the data transfer.

One typical application of standalone data logger is in drilling riser (see Figure 1-1 [4]). The data logger is used to record the riser displacements as the result of vortex induced vibration (VIV) in the offshore oil drilling [5]. For example, 2H standalone data logger can be used as a monitoring device which contains sensors, logger, batteries and memory disk Figure 1-2.

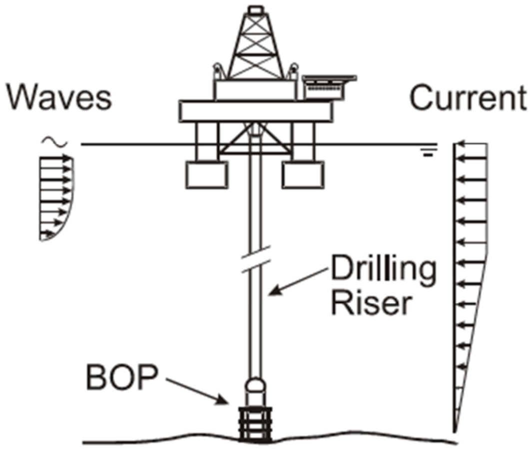


Figure 1-1. A typical floating drilling platform connected to the wellhead through a pipeline called “drilling riser” [4]. A Blow-Out-Preventer (BOP) is installed above the wellhead. BOP is a safety valve to prevent uncontrolled release of the fluid or the gas.

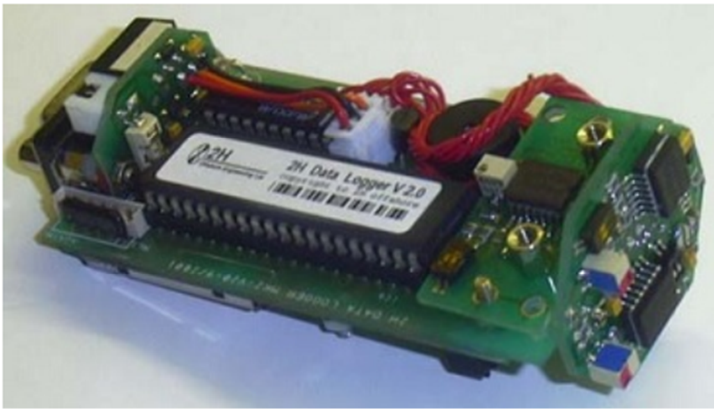


Figure 1-2. 2H Data Logger [5]. A standalone data logger is a data-capturing device that can automatically log data and store it in its on-board memory for a later processing of data stored in memory (e.g. a computer).

Another traditional method is the wired sensor network. The wired sensor networks are used for real-time monitoring or data collection [6]. Wires are used to provide electrical power to sensors along with the data transfer. This is a significant advantage of the wired sensor networks. However, there are some disadvantages as well. Foremost disadvantage is the wire damage which may completely cut off the communication or partially disrupt the communication. The cost of the wire is another significant disadvantage, especially in case of the fiber optic cable. For example, for the monitoring of the long pipelines, the cost of the cable becomes very high. Expansion of the network is also not an easy task for wired networks. Due to these disadvantages of the traditional underwater sensor networks, there has been growing interest in deploying the wireless sensor networks in oceans. There are many applications of the UWSNs. For example, military surveillance, environmental monitoring, early warning systems, structural monitoring [7], micro-habitat monitoring [8].

Although, the performance metrics of TWSN and UWSN are the same, the underwater environment is very much different form the terrestrial environment in many aspects. Electromagnetic waves cannot travel more than few meters in water. Absorption rate of electromagnetic energy in sea water is about $45\sqrt{f}$ dB (f in Hz) [9]. Lloret et al. [10] performed a test to find the distance traveled by radio waves in water. The test was performed in a swimming pool, using 2.432 GHz signal. The optimum performance of the transmitted signal was recorded at 16 cm away from the transmitter. The results of this experiment clearly show that the electromagnetic waves have very high absorption rate in water. Although, low frequency (3-300 Hz) electromagnetic waves can travel some distance, the large antenna requirement makes them impractical. Due to these reasons, radio waves are not used for wireless underwater communication. Therefore, instead of radio waves, acoustic waves are used for underwater communication.

Acoustic waves are found to be more suitable for underwater communication as they can travel much longer distance compared to radio waves. The underwater acoustic communication system was first used for the communication between the submarines around the end of Second World War [11]. It used single-side band AM in 8-11 kHz band. Although acoustic waves travel long distance, they pose new challenges for UWSN communication.

1.5 UWSN Applications

There are many applications of UWSN in the domains of military surveillance, disaster monitoring, environmental monitoring, and water sports. Natural disasters like tsunami and man-made disasters have urged the need of continuous monitoring to get early warning and avoid the damage due such disasters. Tohoku tsunami, in 2004, caused the damage of USD 150 billion [12]. The leaks in underwater oil and gas pipelines require visual inspection before to choose the repair method. If the leak is due to the pinholes, then the mechanical pressure clamps may be used to arrest the leak. But for waters where the visibility is quite poor, the repair process becomes exceedingly difficult. To overcome this problem different kind of sensors like hydrophone sensors or hydrocarbon detector can be used [13]. The applications of UWSN are discussed in more detail below.

1.5.1 Military Applications

UWSNs can be very useful to detect any intrusion from the enemies beyond the water boundaries. The intruding object may be as large as a submarine or as small as a Bluefin-21, which is a 16 ft long unmanned underwater vehicle (also known as Autonomous Underwater Vehicle, AUV). Generic Littoral Interoperable Network Technology (GLINT), known as GLINT10 [14], is a trial based undersea surveillance system based on AUVs. It is a submarine monitoring system. The surveillance system used Ocean Explore (OEX) AUVs (Figure 1-3) which are 4.3 m long.

A project like GLINT10 was tested in Norway in 2011. The project was called UAN11 (Underwater Acoustic Network 11) [15]. UAN11 tested the surveillance of offshore and coastline infrastructures. UAN11 used eFolaga AUVs which were equipped with KM modem as shown in Figure 1-4.

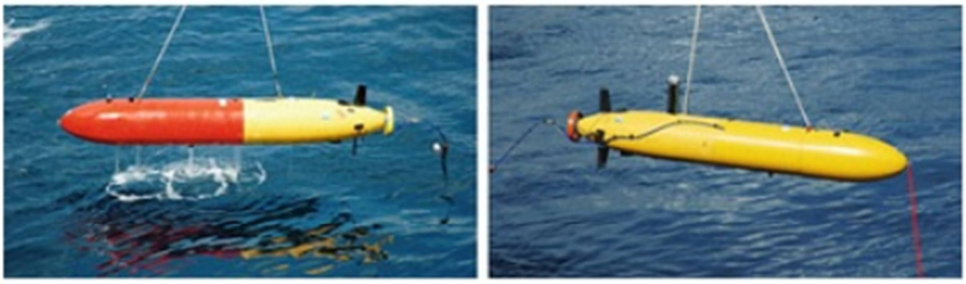


Figure 1-3. OEX AUVs [14]. These are surveillance AUVs equipped with Doppler Velocity Log (DVL) for navigation and acoustic modem.

Figure 1-4. eFolaga AUVs Equipped with KM Modems [15]. These are surveillance AUVs controlled from a remote station to respond against any intrusion.

1.5.2 Disaster Monitoring

Monitoring of seismic activity for tsunami early warning is one of the major applications of the UWSN. Seismic sensors are deployed at the bottom of the sea, which send the data to the onshore base station through a gateway on the sea surface. The German-Indonesian Tsunami Early Warning System (GITEWS) [16] is such a system installed in Indian ocean near by Indonesian coastal area . It is comprised of several base stations along the coastal area of Indonesia (Figure 1-5).

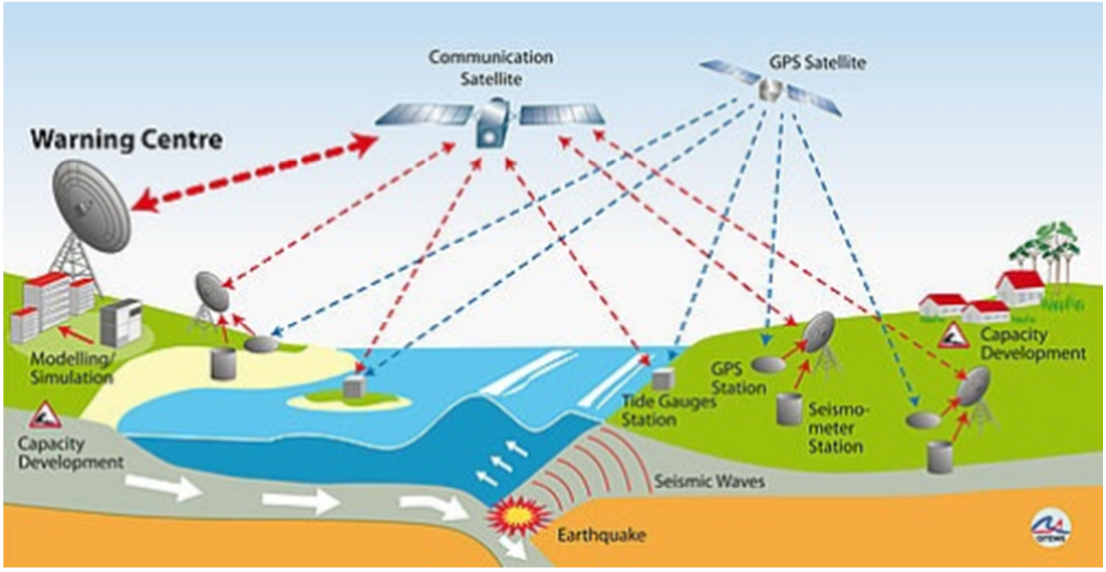


Figure 1-5. GITEWS Network Concept [16]. The network shows the instruments and their interaction to get the prompt Tsunami warning. Seismometer detects the earthquake's strength and location. GPS is used along with seismological data to characterize the earthquake fracture. GPS tide gauges detect the change in the water level due to Tsunami.

National Oceanic and Atmospheric Administration (NOAA), USA, deployed Deep ocean Assessment and Reporting of Tsunamis (DART) system (Figure 1-6) in 2001 [17] as tsunami early warning system. Initially the network had 6 buoys which were eventually extended to 39 buoys by 2008.

1.5.3 Oceans and Rivers Environmental Monitoring

Ocean plants play a key role in marine ecosystem. Oceans plants generate more than half of the oxygen present in the atmosphere and absorb carbon dioxide from the atmosphere. The marine plants like phytoplankton, algal plankton and kelp produce oxygen. Kelp also supplies food for ocean animals and used by human beings for things like ice-cream and toothpaste. Like kelp, phytoplankton is also used as food by ocean animals, even large animals, like whales. The health of all organism in the ocean depends on phytoplankton [18]. Concentration of phytoplankton should be kept balanced in the ocean ecosystem. An imbalance concentration of phytoplankton is harmful for the marine life. Marine plants live in euphotic zone of the ocean which is the layer closer to sea or lake water surface which receives sufficient light for the photosynthesis process to occur. The productivity of plankton decreases with insufficient sunlight. Deficiency of plankton productivity causes deficiency of oxygen in the ocean which affects the marine life detrimentally. On the other hand, increase in the productivity of algae and plankton is also harmful. Increase in the productivity of algae and plankton is caused by the increase of the nutrients in the ocean. An increase of the nutrients, like phosphorous, nitrogen and iron, is called eutrophication. According to ILEC's survey of the world lakes [19] 54% of Asian lakes, 53% of European lakes, 48% of North American lakes, 41% of South American lakes and 28% of African lakes are affected by eutrophication. Overproduction of phytoplankton because of eutrophication can cause red tides which may contain harmful toxins that can affect humans and marine life. Therefore, monitoring of phytoplankton concentration is done by the chlorophyll sensors and algae concentration is monitored by blue-green algae sensors.

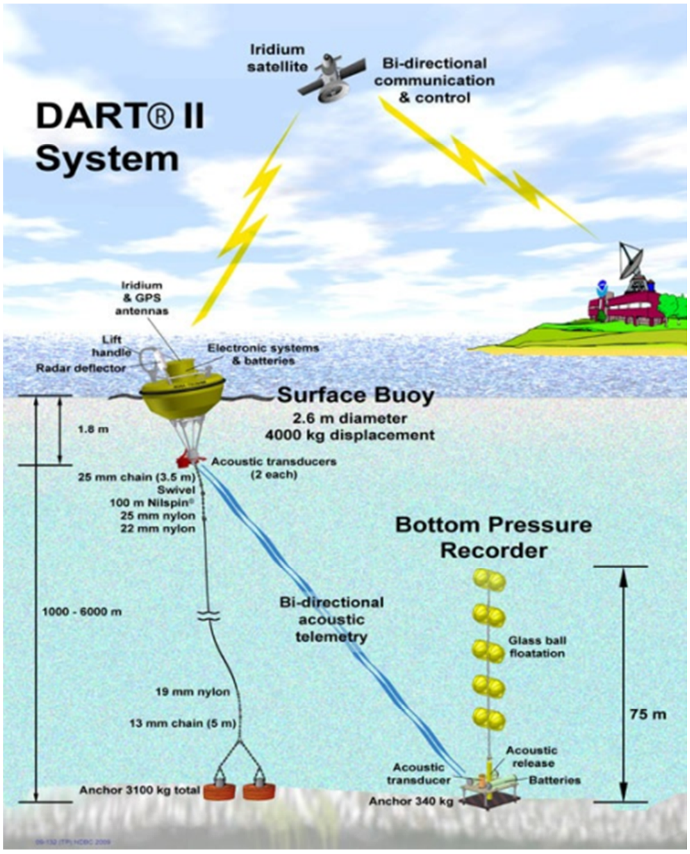


Figure 1-6. DART Network [17]. Bottom Pressure Recorder (BPR) measures temperature and pressure after every 15 sec. There is a two-way communication between BPR and Tsunami Warning Center (TWC). If a Tsunami is expected than TWC sets BPR to event mode to detect Tsunami immediately.

The Acoustic Communication network for Monitoring of Environment (ACMENet) [20] was a marine environment monitoring project to measure the current profile near Westerschelde harbor, Netherlands. The accurate knowledge of current profile is important to guide the ships to the harbor. ACMENet is based on polling mechanism in which a sink polls the nodes to get data scheduler based on TDMA access method. However, the test deployment was not a complete success due to the node's failures. Some sensor nodes were buried in the sand soon after the deployment.

Monitoring of temperature and luminosity on the Australian Coral Reef at Moreton Bay, Australia, is carried out by a WSN [21]. The nodes are equipped with the sensors to get the data and solar panel is used to recharge the node's batteries. The nodes floating on the sea surface are shown in Figure 1.7. The gateway also floats on the sea surface and has two antennas to communicate with nodes and the base station (**¡Error! No se encuentra el origen de la referencia.**).

Figure 1-7. Sensor Node [21]. It is composed of control unit and data processing, signal acquisition and conditioning, local transmission radio, energy-harvesting mechanisms, and energy storage

Figure 1-8. Gateway [21]. Gateway is same as the sensor node with additional long-range RF communication capability to communicate with the base station.

Zhang et al. [22] used marine vehicles in the South China Sea to monitor the microscale and meso-scales dynamic processes. The main objective was to combine the motion control of the marine vehicles with their trajectory path planning to get more samples of the data. Unmanned Surface Vehicles (USVs), Autonomous Underwater Vehicles (AUVs) and Autonomous Underwater Gliders (AUGs) were used to collect data on surface of the sea, at different water layers (see Figure 1-9). Mixed Integer Linear Programming (MILP) defines the trajectories of the vehicles which provide the uniform distribution of the vehicles.

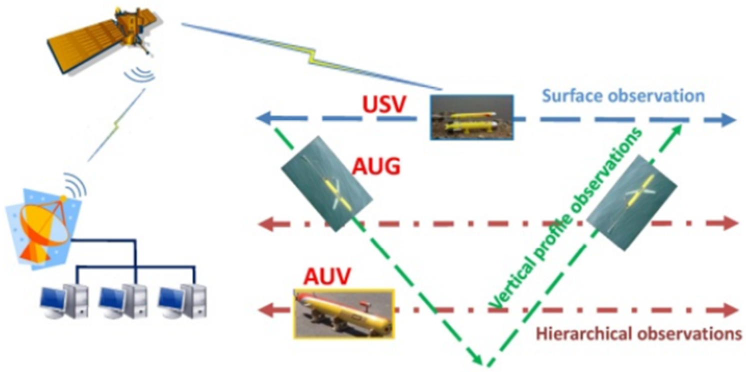


Figure 1-9. Motion Characteristics of USVs, AUVs, and AUGs [22]. USVs are suitable for observations on the sea surface. Autonomous Underwater Vehicles (AUVs) are used for observations at different layers under the sea surface. Autonomous Underwater Gliders (AUGs) glide up and down underwater using hydrodynamic force from the wing. USV carries out ocean surface observations and transfers data between AUG, AUV and OSDSS

1.6 Behavior of Underwater Acoustic Waves

In order to understand the propagation behavior of acoustic waves in water we briefly describe some characteristics like velocity, fading, transmission loss, noise, and bandwidth capacity of the acoustic waves.

1.6.1 Sound Velocity

Sound is originated by mechanical perturbation or oscillation of an object. When an object oscillates, it causes change in pressure in its surrounding environment. For example, striking a drum surface with a drumstick will cause it to vibrate. This vibration of the drum surface will compress and decompress the air which travels through the air and creates the sound wave.

The propagation velocity of an acoustic wave depends on the density ρ and elasticity modulus E (which is the inverse of compressibility). The sound velocity increases with the depth in sea water due to hydrostatic pressure. The variation in the bulk modulus B causes the increase in the sound velocity. This increases in velocity with respect to the depth is almost linear. Typical value is around 0.017 m/s/meter increase in depth. The Leroy [23] formula for hydrostatic pressure is:

$$P = 1.0052405(1+5.28x10^{-3}\sin\phi)z + 2.36x10^{-6}z^{2} \qquad 1.1$$

where φ is the latitude in degrees and z is the depth in meters.

There are several models to estimate sound velocity. Medwin [24] is a simple sound estimation model which limited 1000 meters depth. Medwin model is as given below:

$$c(t, z, S) = 1499.2 + 4.6\,T - 0.055\,T^{2} + 0.00029\,T^{3} + (1.34-0.01\,T)(S-35) + 0.016D, \qquad 1.2$$

where c is the velocity of the sound, T is from 0°C to 35°C, z is 0-45 ppt and D is 0-100 m. Chen and Millero [25] model is more complex but considered more accurate as it is adopted by the UNESCO as a reference model. Chen and Millero model is as given below:

$$U(S,t,p) = C_{w}(t,p) + A(t,p)S + B(t,p)\,S^{2/3} + D(t,p)\,S^{2} \qquad 1.3$$

where S is salinity (practical salinity unit), t is temperature (C) and p is pressure (decibar)

$$C_{w}(t,p) = C_{0} + C_{1}p + C_{2}p^{2} + C_{3}p^{3} \qquad 1.4$$

$$C_{0} = 1402.388 + 5.03711t + 5.80852x10^{-2}t^{2} + 3.3420x10^{-3}\,t^{3} - 1.478x10^{-6}\,t^{4} + 3.1464x10^{-9}\,t^{5} \qquad 1.5$$

$$C_{1} = 0.153563 + 6.8982x10^{-4}t - 8.1788x10^{-6}\,t^{2} + 1.36721x10^{-7}\,t^{3} - 6.1185 + 1.362x10^{-10}\,t^{4} \qquad 1.6$$

$$C_{2} = 3.126x10^{-5} - 1.7107x10^{-6}\,t + 2.5974x10^{-8}\,t^{2} - 2.5335x10^{-10}\,t^{3} + 1.0405x10^{-12}\,t^{4} \qquad 1.7$$

$$C_{3} = -9.7729x10^{-9} + 3.8504x10^{-10}\,t - 2.3643x10^{-12}\,t^{2} \qquad 1.8$$

$$A\,(t,P) = A_{0} + A_{1}p + A_{2}p^{2} + A_{3}p^{3} \qquad 1.9$$

$$A_{0} = 1.389 - 1.262x10^{-2}\,t + 7.164x10^{-5}\,t^{2} + 2006x10^{-6}\,t^{3} - 3.21x10^{-8}\,t^{4} \qquad 1.10$$

$$A_{1} = 9.4742x10^{-5} - 1.258x10^{-5}\,t - 6.4885x10^{-8}\,t^{2} + 1.0507x10^{-8}\,t^{3} - 2.0122x10^{-10}\,t^{4} \qquad 1.11$$

$$A_{2} = -3.9064x10^{-7} + 9.1041x10^{-9}t - 1.6002x10^{-10}\,t^{2} + 7.988x10^{-12}\,t^{3} \qquad 1.12$$

$$A_{3} = 1.1x10^{-10} + 6.649x10^{-12}\,t - 3.389x^{10-13}\,t^{2} \qquad 1.13$$

$$B = B_0 + B_1 p \quad 1.14$$

$$B_0 = -1.922x10^{-2} – 4.42x10^{-5}\ t \quad 1.15$$

$$B_1 = 7.3637x10^{-5} + 1.7945x10^{-7}\ t \quad 1.16$$

$$D(t,p) = D_0 + D_1 p \quad 1.17$$

$$D_0 = 1.727x10^{-3} \quad 1.18$$

$$D_1 = – 7.9836x10^{-6} \quad 1.19$$

The sea water density is about 1,030 kg/m^3. In sea water, velocity of the acoustic wave is between 1450 m/s and 1550 m/s. The underwater acoustic waves velocity and density depend on pressure, temperature, and salinity. Usually, the underwater acoustic waves velocity is considered as 1500 m/s. The temperature of the sea water decreases from the surface to the seabed. After certain depth the temperature decreases very slowly in the sea water. In open sea that depth is about 1000 m and in Mediterranean sea it is about 100-200 m. Typical value of salinity is 35 practical salinity units (PSU, or g/kg) in open sea, 38 PSU in closed sea like Mediterranean sea and 14 PSU in Baltic sea due to freshwater input. Normally, salinity has little effect on sound velocity with change in depth but may become significant at estuaries.

The characteristic acoustical impedance ρc, where ρ is water density and c is wave velocity, relates the acoustic pressure level to the corresponding motion of particles. The SI unit for acoustic impedance is Pa·s/m, commonly known as Rayleigh, abbreviated Rayl. Acoustic pressure level is much higher in high impedance medium like water (ρc = 1.5 MRayl) compared to low impedance medium like air (ρc 400 Rayl).

1.6.2 Multipath Fading

The underwater acoustic waves experience sever multipath fading due to reflections off the sea bottom and sea surface and scattering from non-homogeneity in water column (Figure 1-10 [26]). The multipath fading causes interference at the receiver.

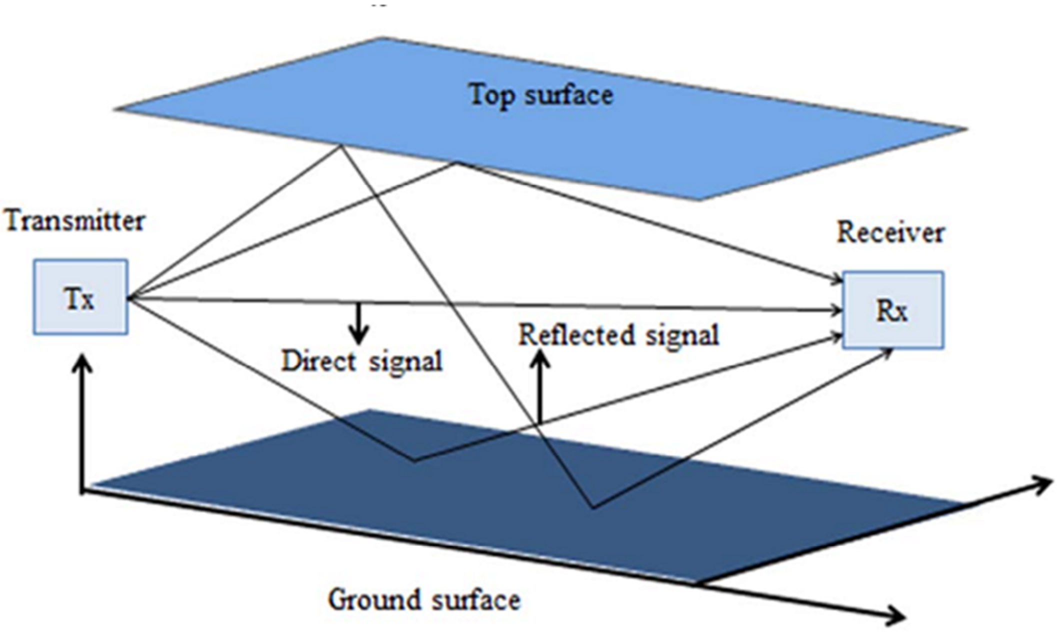


Figure 1-10. Multipath Fading in Underwater Acoustic Communication [26]. There is direct signal path between the transmitter and receiver along with reflected signals from the bottom and surface of the sea.

Delay spread causes intersymbol interference (ISI) which corrupts the received data. When the delay spread is larger than the symbol period, the ISI occurs. Since the delay spread depends on the environment, it cannot be controlled. However, symbol period can be increased to overcome the ISI. Sound velocity profile and the geometry of the channel directly affect the reflection and refraction pattern of the sound waves. The channel impulse response of the water channel, especially sea water, varies over time due to movement of the channel. The currents and tides cause the channel movement. Coherence time and Doppler spread describe the channel's time-varying behavior.

1.6.3 Transmission Loss

The bandwidth of the acoustic signal is also very limited due to transmission loss. The transmission loss is caused by attenuation and spreading. The key factor in the attenuation is absorption which is the conversion of acoustic energy in to heat. The absorption of the signal is highly frequency dependent and is strongly influenced by the depth of the sea. Water bubbles not only absorb and scatter the acoustic signals, but they also change the speed of the signal few meters below the sea surface [27]. The absorption loss puts limit on the useable bandwidth because the absorption loss increases with frequency as well as with distance. Hence, higher bandwidth can be achieved at shorter distance [28]. Assume that the distance between two communicating nodes, A and B, is 10 km. Data is sent from A to B at the data rate 10 kHz. Now assume that there are 10 nodes at distance of 1 km between A and B which can relay the data from A to B then, the data rate may increase to 20 kHz.

Attenuation or path loss in the acoustic wave depends on the distance and the frequency. Attenuation of an acoustic signal is given as [29]:

$$A(l, f) = l^k \, a(f)^l, \qquad 1.20$$

where k is the spreading factor, and $a(f)$ is the absorption coefficient. The acoustic path loss expressed in terms of dB is,

$$10 \log A(l, f) = 10 \log (l^k . a(f)^l)$$

$$10 \log A(l, f) = 10 \log l^k + 10 \log a(f)^l$$

$$10 \log A(l, f) = k \, 10 \log l + l \, 10 \log a(f) \qquad 1.21$$

The first term in the right side of the above equation represents spreading loss and the second term represents the absorption loss. The absorption coefficient $a(f)$ is calculated using Thorp's empirical formula as shown in Equation **¡Error! No se encuentra el origen de la referencia.**.

$$10 \, logA(l, f) = 0.11 \frac{f^2}{1 + f^2} + 44 \frac{f^2}{4100 + f} + 2.75 \times 10^{-4} f^2 + 0.003 \qquad 1.22$$

Figure 1-11 shows the absorption coefficient with respect to frequency. It is clear from the figure the absorption coefficient rises rapidly with increase in the frequency and hence, imposes limit on the useable frequency.

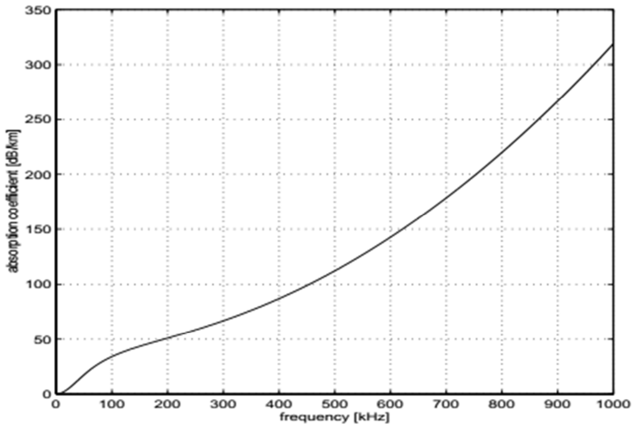


Figure 1-11. Absorption Coefficient a(f) vs. Frequency [29]. The graph shows that the absorption coefficient increases with the increase in the frequency

The acoustic wave spreads more and more and the intensity of the wave decreases inversely proportional to the wave surface. The spreading of the beamwidth of the signal can be classified as cylindrical spreading and spherical spreading.

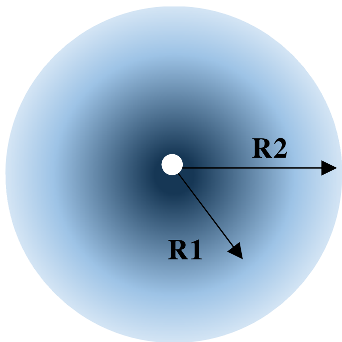


Figure 1-12 Spherical Spreading: The acoustic intensity decreases with distance from the source. [30].

The intensity ratio between the points A and B at the distance R1 and R2, as shown in Figure 1-12 is given by [30]

$$\frac{I_2}{I_1}=\frac{\Sigma_1}{\Sigma_2}=\frac{4\pi R_1^2}{4\pi R_2^2}=\left(\frac{R_1}{R_2}\right)^2 \qquad 1.23$$

The transmission loss due to cylindrical spreading is proportional to 1/R and spherical spreading is proportional to $1/R^k$, where R is the distance between the source and the receiver. The value of k = 1 for cylindrical spreading and k = 2 for spherical spreading. However, in shallow waters the spreading is partially cylindrical and partially spherical, and k = 1.5 [29].

1.6.4 Channel Noise

The different channel noise sources affect the different bands of acoustic waves. There are four main sources of channel noise. The Equations **¡Error! No se encuentra el origen de la referencia.**, 1.25, 1.26 and 1.27 show the channel noises due to different sources [28]. Below 10 Hz frequency acoustic signals, the turbulence noise has the greater effect. The sea turbulence noise (TBN) is defined as:

$$TBN = 17 - 30 \log f, \qquad 1.24$$

where f is the frequency in Hz. Shipping Noise (SHPN) affects the acoustic signals from 10 Hz to 100 Hz.

$$SHPN = 40 + 20(s - 0.5) + 26\log f - 60\log (f + 0.03) \qquad 1.25$$

The roughness of the sea due to surface winds affects the signals between 100 Hz to 100,000 Hz. The roughness noise (RN) of the sea is defined as:

$$RN = 50 + 7.5/2 + 20\log f - 40\log (f + 0.4), \qquad 1.26$$

where w is the wind speed in m/s. Above 100,000 Hz the thermal noise (THN) has the greater effect on the signal. The thermal noise is defined as:

$$THN = -15 + 20\log f \qquad 1.27$$

Total Noise, N is defined as:

$$NL = 10\log(10^{\frac{TBN}{10}} + 10^{\frac{SHPN}{10}} + 10^{\frac{RN}{10}} + 10^{\frac{THN}{10}}) \qquad 1.28$$

1.6.5 Relationship between the Bandwidth Capacity and the Transmission Distance

The product of the attenuation and noise is inversely proportional to signal-to-noise ratio (SNR). The SNR depends on the frequency of the acoustic signal. Figure 1-13 [29] shows that there is a large SNR for each distance at a certain frequency. For example, for transmission up to 5 km the largest SNR that can be achieved is about -90 dB at around 9 kHz. This is called the optimal frequency f_{op}.

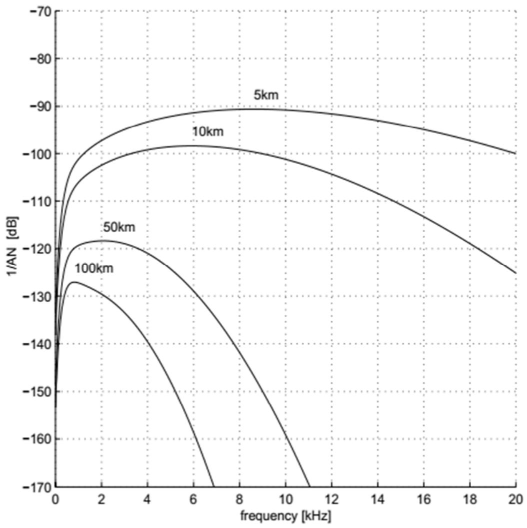


Figure 1-13. Frequency vs SNR [29]. It shows the largest SNR that can be achieved for a given distance at certain frequency

The channel capacity is given as [28]:

$$C(l)=\int_{B(l)} \log_2 \left[\frac{K_l}{A(l,f)N(f)}\right] df, \quad 1.29$$

where K_l is a constant defined as:

$$K_l = S_l(f) + A(l,f)N(f) \quad 1.30$$

and where $S_l(f)$ is the power spectral density (p.s.d) of the transmitted signal.

1.7 UWSN Architecture

Underwater wireless sensor network architecture can be divided in to two categories: (a) two – dimensional (b) three – dimensional.

1.7.1 Two-Dimensional Architecture

Akyildiz [31] defines the two dimensional network as a network in which the nodes are anchored to the bottom of the sea. Such a network is useful in the shallow sea, when the area of interest is only the bottom of the sea. One of the possible deployments of such network is shown in Figure 1-14

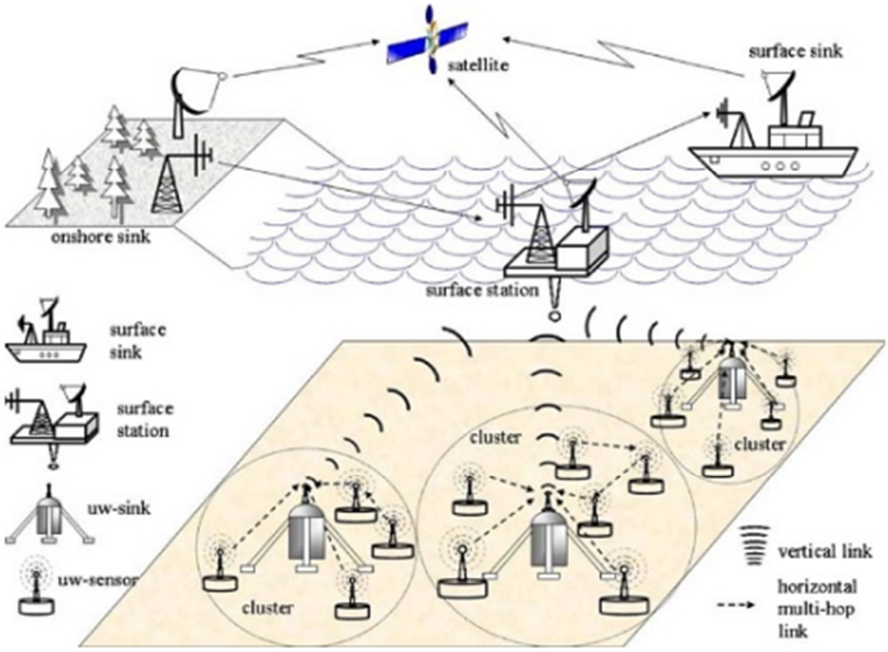


Figure 1-14. Two Dimensional UWSN Architecture [31]. The sensor nodes are fixed to the bottom of the sea. The uw-sink has bigger energy storage and longer communication range to forward the data of the nodes in a cluster to the surface station.

There is a node at the surface of the sea which communicates with the underwater nodes using acoustic waves and the onshore station using electromagnetic waves. This surface node is more commonly known as sink or *gateway*. The sensor nodes will relay the data to the sink using more powerful intermediate underwater sinks (uw-sinks). Either the sensor nodes can communicate with the uw-sinks directly or using hop-by-hop approach. The nodes connected to a uw-sink will form a cluster and where uw-sink acts as a cluster head. As with any cluster-based network, these cluster heads become single point of failure and need to be having a large power source to achieve long network life. Some method of routing the data within the clusters may also be required.

1.7.2 Three-Dimensional Architecture

When the area of interest is not just the bottom of the sea, but the data is to be collected from the surface to the bottom of the sea at multiple depths, then nodes are deployed at different depth levels. The nodes can be anchored to the bottom of the sea with long rope or chain to be deployed at different lengths (Figure 1-15). Inflatable buoys can also be used to adjust the depth of the nodes (Figure 1-16).

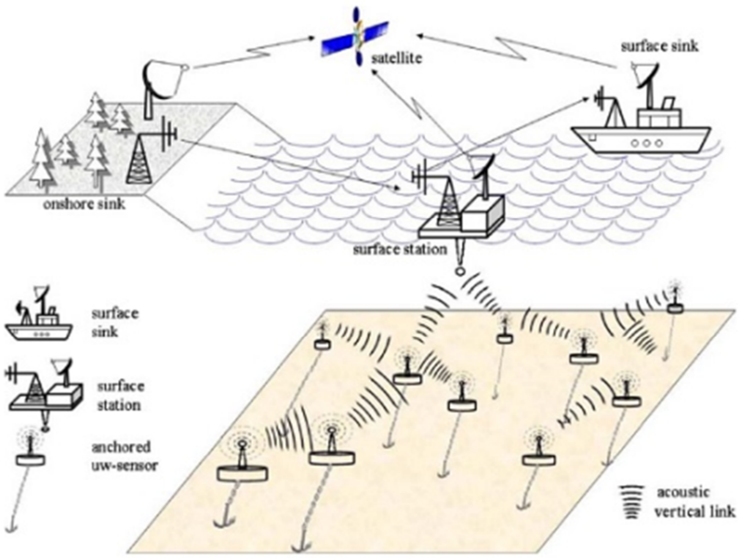


Figure 1-15. Three Dimensional Anchored Node [31]. The sensor nodes are anchored to the sea bottom and can move manily horizontaly and little bit vertically. The forward the data to the surface stattion either directly or via other sensor nodes.

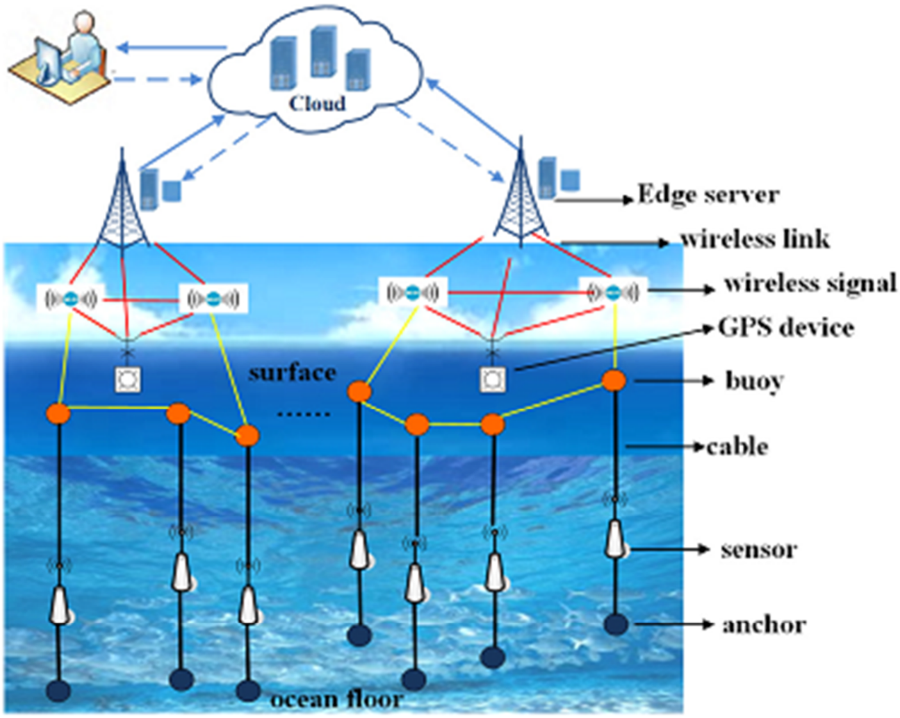


Figure 1-16. Three Dimensional Buoy Supported Sensor Nodes [32]. The sensor nodes are suspended using buoys and anchors at the bottom of the sensors are used to keep their horizontal movement minimum.

Other than the two categories of the UWSN architectures mentioned above, one more possible method of data collection underwater and relaying it to the surface station is by autonomous underwater vehicles (AUVs). AUVs are also known as unmanned underwater vehicles. AUVs move around at the pre-defined path using stored computer programs. In contrast to AUVs, remote underwater vehicles (ROVs) are remotely navigated with the help of cables and camera. [33] is a distributed data gathering protocol which uses AUVs for data gathering (Figure 1-17). The sensors may also use radio links if the AUV or ROV is very close to the data transmitting node and the amount of the data is too large.

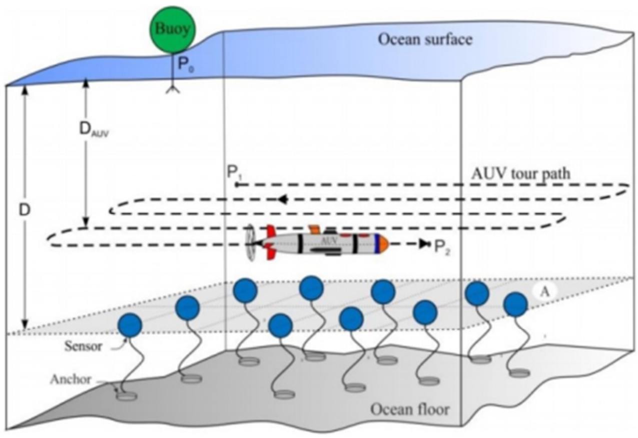


Figure 1-17. AUV Based UWSN Network [33]. AUV is used to collect the data from the sensor nodes which are anchored to the bottom of the sea. When AUV finishes collecting data it transfers the data to the surface station.

1.8 UWSN Issues

In this section we discuss some distinctive issues pertinent to UWSN.

1.8.1 Time Synchronization and Localization

Time synchronization and localization are required for medium access and routing protocols. Time synchronization and localization are also needed in situations where the time and location of the data collection point are also recorded along with the data. For example, in case of intrusion detection system, if an intrusion is detected by a sensor node, then the time and location of the intrusion is also required to be sent by the sensor node. For that, the sensor node must have time synchronization and location information.

For terrestrial networks, time synchronization and localization are easily achieved by the global navigation satellite system (GNSS) embedded in the senor nodes. However, UWSN the underwater nodes cannot communicate with GNSS satellite directly because the underwater nodes use acoustic waves whereas GNSS satellites use electromagnetic waves. In addition to that, large propagation delay and limited movement of nodes also make the time synchronization a challenging task. Time synchronization is also difficult to achieve due to variable speed of acoustic waves which varies with temperature and pressure. The time synchronization signaling overhead should be minimized for UWSNs due to low data rate. Therefore, the time synchronization and localization protocols used for terrestrial wireless networks cannot be readily used and new a mechanism is required for UWSNs.

1.8.2 Deployment and Replacement of Sensor Nodes

Unlike most terrestrial WSNs, the deployment of UWSN nodes is difficult, time consuming and costly, especially in deep waters. The active life of the sensor nodes becomes more subtle in UWSNs because the replacement of the nodes is very difficult and expensive. The senor nodes are usually placed in strong casing to protect them from the harsh environment of the sea which makes them bulky and expensive. The sensor nodes are more prone to failures due to fouling and corrosion [34].

1.9 Thesis objectives

The main objective of this research is to develop a UWSN routing protocol which is self-organized, resilient, and having low routing delay for time-critical applications. Time-critical applications need smallest possible end-to-end packet delivery delay. There are many sources of delay on a data network such propagation delay, packetization delay and routing delay. The routing delay becomes a significant source of packet delay in data networks which are large and produce heavy data traffic. The specific objectives are given below.

- Thoroughly study and classify UWSN routing protocols to identify the issues pertinent to the packet routing.
- Develop a minimum routing delay protocol for a UWSN network having a predefined topology.
- Develop a minimum routing delay protocol for a UWSN network in which the nodes are not deployed in a specific topology.
- Simulate the routing protocols to estimate the time taken by a network to form the routing paths.

1.10 Conclusion

In this chapter we have discussed UWSN applications, network architectures, and issues. EM waves cannot be used for underwater communication due to high absorption rate. Therefore, acoustic waves are used in UWSNs, but they travel very slow compare to EM waves. Typical speed of acoustic waves in the water is 1500 m/s. WSN protocols are not designed to account for such a high propagation delay. Therefore, new protocols are required for UWSN networks. UWSN have wide applications such as surveillance, disaster monitoring and environmental monitoring.

The velocity of acoustic waves in the water depends on the density and elasticity modulus of the medium. The sound velocity increases as it travels deeper into the water. Underwater acoustic wave impairments are multipath fading, transmission loss and channel noise.

The multipath fading is sever in sea waters and causes signals interference at the receiver. The transmission loss is caused by the attenuation and spreading. The attenuation increases as the wave propagates farther away from the source. The bandwidth depends on the attenuation of the signal and the center frequency. Hence, the bandwidth decreases as the propagation distance and the center frequency increase. In sea waters, there are four main sources of the noise namely turbulence noise, shipping noise, roughness noise and thermal noise.

The sensor in UWSN network can be deployed in two ways: 1) fixed at the bottom of the sea 2) floating at different sea level. When sensor nodes are fixed at the bottom of the sea, we call it 2D architecture and when they are floating, we call it 3D networks. 2D networks are used for shallow waters whereas 3D networks are used for deep waters. Sensor nodes in 3D networks have limited mobility due to water currents. Some UWSN networks use AUVs to collect the data from the senor nodes and relay to the sink node.

Challenges in achieving the time synchronization and location of the nodes are UWSN specific issues. The medium access and routing protocols using timestamp or location of the node are prone to failure due to errors in time synchronization and localization.

1.11 Thesis Organization

In Chapter 2 we have analyzed the existing routing protocols for their strengths and weaknesses to establish the need of a new routing protocol. In Chapter 3 we have described the details of our proposed routing protocol for radial networks. In Chapter 4 we described the details of a routing protocol for randomly deployed sensor nodes. Chapter 5 is the conclusion our research work described in this thesis.

Chapter 2
State of the Art

2.1 Introduction

In this chapter, we will review the existing routing protocols for UWSN. This will help us to understand the routing issues in UWSN in depth. The routing protocols can be classified in two broad categories, namely, proactive, and reactive. In proactive protocols, the forwarding nodes already know the next node to which the packet has to be forwarded, which reduces the packet forwarding time. Proactive protocols are more suitable for delay sensitive data. However, they have some disadvantages as well. They need to update the routing tables whenever the network topology changes, or a node failure occurs. Scalability is another problem of proactive protocols [35].

In reactive protocols a node finds the next hop to forward the data when it receives a packet to be forwarded [36]–[39]. This causes delay and makes this type of protocol unsuitable for delay sensitive data. However, periodic updates are not required, and the network is easily scalable.

The UWSN routing protocols can also be classified as location-based or non-location based. Location-based routing protocols choose the next forwarding node on the basis of nodes location in the network. In most cases, to determine the location of an underwater node, some additional sensing device, like a pressure sensor, is required. For example, the next node can be chosen based on its distance to the destination or based on depth measurements. The additional sensor device causes more energy consumption, which is an undesired characteristic for UWSNs. In addition to that, determining the location of a node accurately in 3D UWSNs is difficult because of continuous mobility of the sensor nodes.

The routing protocols can be classified as: location free protocols, location-based protocols, opportunistic protocols, cluster-based protocols, energy efficient protocols, mobility-based protocols and reliable data deliver protocols. From Section 2.2 to Section 2.8, we have classified and briefly discussed some routing protocols on the basis of above mentioned classifications.

2.2 Location Free Protocols

Network Layer Protocol for UANs (NLPU) [40] is a self-configured, proactive protocol based on the concept of full-duplex channel utilization. The gateway is the master node and manages the topology of the network and establishes the path through the transmission of a topology discovery probe. The probe traverses the network and each node adds its identification in the probe and relay it to the next node. The protocol uses sliding window link layer protocol to increase the data delivery reliability. The probe also assigns the channels to the nodes as it traverses from the gateway node outward. However, conflict is possible if the nodes select the same channel. This can be resolved by the parent node by assigning the channel to one of the conflicting nodes and asking other conflicting nodes to choose another channel. Once the probe completely traverses the network, completion notice is sent back to the gateway. The protocol heavily relies on full-duplex communication for efficient utilization of channel. However, dividing total channel bandwidth of 5kHz (for 100 m propagation range) [41] into 7-12 sub channels gives 0.714 kHz - 0.416 kHz bandwidth respectively for each channel. Assuming binary signals are used, then according to Nyquist maximum channel capacity formula for noiseless channel can be 1428 bps - 832 bps respectively.

Location Free Link State Routing (LFLSR) [42] addresses the problem of communication voids. Every node selects the next hop based on three metrics, namely hop count, path quality, and depth (pressure). A beacon message is broadcast from the sink node. The nodes which receive the beacon message update their path quality and hop count state, and broadcast a new beacon message. To select the next hop the forwarding node selects the one-hop neighbor. In case of more than two one-hop neighbors, the forwarding node selects the highest path quality neighbor. If multiple nodes have equal path quality too, then the forwarding node selects the neighbor with the lowest pressure. This protocol requires a pressure measurement device, which consumes more energy.

Localization-Free Interference and Energy Holes Minimization (LF-IEHM) [43] is based on nodes depth, the distance between the sender and the neighbor and adaptive transmission range to overcome the problem of energy holes. Energy hole is created when single node or multiple node die due to drain of energy. The energy hole disconnects the communication between the source and destination node if dead nodes are there in routing path. LF-IEHM increases the transmission range to overcome the problem of energy hole. If a node does not get any response, then it increases the transmission range to reach a live node in its neighbor. There are few issues not addressed in this protocol. All the nodes maintain a table of pressure levels (depths) and IDs of neighbors and later exchange those tables via broadcasting. Since there is no mechanism for the broadcasting of these packets containing table, collision may occur. This protocol also requires a depth measurement device, which consumes more energy.

2.3 Location Based Protocols

Vector Based Forwarding (VBF) is a location-based protocol [44]. It introduced the concept of virtual pipes. A virtual pipe is a set of selected nodes from the source to the sink which can possibly forward the packets. Multiple nodes within the virtual pipe forward the data packets to avoid the issues of packet losses and node failures. However, in case of networks with low density of nodes, voids may exist, which decreases the delivery ratio. The power consumption is high due to three-way handshaking. To solve the problem of voids in VBF, Hop-by-Hop VBF (HH-VBF) [45] proposes a virtual pipe from one node to another rather than from source to destination. This way each forwarding node makes a new virtual pipe. The direction of the virtual pipe depends on the location of the forwarding node.

However, signaling overhead of HH-VBF is higher than VBF, due to hop-by-hop virtual piping. The Adaptive Hop-by-Hop Vector-Based Forwarding (AHH-VBF) routing protocol [46] is based on the HH-VBF protocol where the radius of the virtual pipeline is defined in a hop by hop basis. AHH-VBF not only changes the direction of the virtual pipeline but also its radius to restrict the forwarding range of the packets. By reducing the radius of the virtual pipe, the reliability of the packet delivery increases in the sparse network and duplicate packets are fewer than in the dense network. Power control is also implemented to decrease the consumption of energy. LB-AGR [47] routes the packets on the basis of level-difference between the neighbor nodes, available power, node density and location. It divides traffic into four categories, upstream towards the sink, downstream to all the nodes, downstream to a specific node and downstream to all nodes in an area. Compared to VBF the packet delivery ratio is poor, but the average energy consumption is significantly better.

Directional Flooding-based Routing (DFR) [36] is a location based routing protocol. It assumes that every node has knowledge about its own location, the location of the next hop node and the location of the destination node. Forwarding nodes are selected on the basis of link quality and angle of arrival. In Focus Beam Routing (FBR) [48] each node has location information of its own and the final destination. The forwarding area is selected based on transmission power. If there is no node present in the forwarding area, it is increased. Sector-based Routing with Destination Location Prediction (SBR-DLP) [49] assumes that every node knows its own location information and the movement of the destination nodes. The sender gets information from the next possible forwarding nodes and decides which node will be the next hop. Pre-planned movement of the destination node makes implementation of SBR-DLP quite limited. In Location-Aware Routing Protocol (LARP) [50] the sink nodes find their location by GNSS and broadcast their location coordinates. The nodes find their location by the reference of at least three sink nodes. When a node wants to send a packet and does not know the location of the destination, it queries the sink nodes. If the receiving sink also does not know the location, then it asks the other sinks; eventually the location of the destination is sent to the sending node. After getting the location of the destination, the sender finds the next hop by broadcasting the location of the destination node. If the destination node is within the broadcast range it replies directly, otherwise no node replies. Next, the sender transmits a “moving direction” packet and the receiving nodes compare their location with the destination node location. If the receiving node finds that they are in the same direction, then it replies to the sender and the packet is sent. Hop-by-Hop Dynamic Addressing Based Routing

(H2-DAB) [51] uses dynamic addresses for the nodes. The addresses of the nodes change according to their positions at different depth levels. The addresses of the nodes closer to the surface sinks are smaller and become larger as the nodes go down towards the bottom.

Self-Organized Underwater Wireless Sensor Network (SOUNET) [52] is a self-organized routing protocol based on nodes levels. The network is divided in hierarchical levels from the sink node (which is the highest level and called 0 level) and moves to the next level, which is the neighbors of the sink, and keeps increasing until it reaches the last node in the network. At the initialization phase, the nodes communicate with each other by a HELLO packet which contains the sender's identification and its hierarchical level. The node that receives the HELLO packet compares its level with the level mentioned in the received packet and if the level is lower or the same then ignores it, otherwise makes itself the child of the HELLO sending node. This protocol requires determining the depth of the nodes to determine their depth. However, there is no mechanism in the protocol for the nodes to determine their depth and hierarchical level. Moreover, there is no mechanism for a node to choose a parent from the multiple parents from the higher level.

PULRP [53] is a reactive protocol which consists of two phases. First phase is called the layering phase which forms the virtual spheres around the sink based on the proximity of the nodes. Second phase is called the communication phase which selects the relay node on the fly. The relay node is selected from each layer such that the distance between them is the maximum and has enough energy to forward the packet.

Area Localization Scheme ALS [54] does not require measurements like Time of Arrival (ToA), Time Difference of Arrival (TDoA) and Angle of Arrival (AoA) to estimate the distance. Three kinds of nodes are used in ALS, namely, reference nodes (anchor nodes), sensor nodes and sink. The reference nodes know their location and send beacon packets which contain the power level at which the beacon was transmitted. The sensors save the list of reference nodes and their respective transmitted power. The sensors use this information to select a reference node and become part of an area covered by a particular reference node.

2.4 Opportunistic Protocols

Opportunistic routing protocols are based on broadcasting or multicasting the data packet to achieved high reliability. However, multiple transmissions of the same packet by many nodes cause significant energy consumption and make them inefficient in terms of energy consumption. The Depth Based Routing protocol (DBR) [55] is based on opportunistic routing. DBR has no mechanism of recovery in case there is no forwarding node present in the direction of the sink node. In DBR, every node ranks the quality of the path towards the sink. Hop-by-hop forwarding is used; the metric to choose the next hop is based on the number of hops. Every node selects a one-hop neighbor in an area to avoid communication void problem. A sender broadcasts the packet along with its depth information and packet sequence number. Upon receiving the packet, single hop neighbor nodes will compare their depth value with the depth value given in the received packet. If the depth value of the neighbor is less than the depth value of the sender, then it will forward the packet. The node closest to the sink will forward the packet first. However, communication void problems may occur if the nodes are very sparse. DBR also requires an additional pressure (depth) measurement device, which additionally consumes limited energy available at the sensor nodes. The Weighting Depth and Forwarding Area Division DBR (WDFAD-DBR) [56] routing protocol, is based on DBR but unlike DBR it does not select the forwarding nodes according to the depth of the current node. WDFAD-DBR considers the depths of expected forwarding nodes to improve the delivery ratio. To reduce traffic for the location update among the neighbor nodes, a location prediction mechanism is used. The forwarding region is also adaptively changed, according to the density of nodes and the channel conditions, to reduce the duplicate packets. Segmented Data Reliable HydroCast [37] is similar to DBR where routing is based on the depth of the nodes but addresses the problem of voids and local maxima present in DBR. In HydroCast each local maximum node maintains a recovery route towards a neighboring node with higher depth than itself. Since it uses pressure sensors to measure the depth, it has a higher energy consumption compared to DBR.

The Multi-Sink Opportunistic Routing Protocol (MSORP) [57] uses a two tier topology. The bottom tier includes the sensor nodes, the mesh nodes and the underwater (UW) sinks and the top tier contains the surface buoy and the monitoring center. Mesh nodes are considered as more powerful than sensor nodes as they have more memory, longer transmission range and higher processing power. Mesh nodes are recharged through underwater vehicles. Sensor nodes send their data to mesh nodes, which aggregate the data and forward it to the sink. It finally forwards the data to the surface buoys. Void-Aware Pressure Routing (VAPR) [58] uses depth information, hop count and sequence numbers to find the next hop and the directional path to the sink. Basically, it is an opportunistic routing protocol. Sink nodes start the localization process by transmitting beacon packets. In order to avoid the voids in the network, the next hop selection is based on the data forwarding direction and the next-hop's data forwarding direction. Efficient Opportunistic Routing Technology (EFFORT) [59] is an opportunistic routing protocol. It is possible that a node fails to receive data over an unreliable wireless link, which requires the sender to retransmit the packet. This retransmission causes an increase in end-to-end delay and energy consumption. In order to reduce retransmissions, opportunistic routing is used. EFFORT uses opportunistic end-to-end cost (OEC) to reduce the of forwarder nodes which increases the network lifetime. The OEC cost is determined on the basis of a forwarding node's residual energy and link reliability. Opportunistic Void Avoidance Routing (OVAR) protocol [60] allows farther away nodes than the sender node from the sink to take part in packet forwarding. However, the nodes closer the sink node will have priority. This helps to avoid the voids or local maxima problem. This transmission method is different from other such opportunistic protocols where only the nodes closer than the sender node to the sink are allowed to forward the packet.

Energy-Efficient and Obstacle-Avoiding Routing (EOAR) protocol [61] uses fuzzy logic to select the forwarding node. The selection is made based on residual energy, angle between the two nodes and the propagation delay. Packets are forwarded based on priority to avoid the collisions. The propagation delay is calculated from the timestamp included in the received packet from the time when the packet was received. This method needs time synchronization between the nodes which is a challenging task for UWSNs due to varying propagation delay and some degree of mobility.

GDGOR-IA and GRMC-SM protocols [62] use location information of the nodes to form the routing path. The routing path is established when the data is to be forwarded which increases the routing delay. The data forwarding uses greedy approach. The network is divided in logical cubes and a set of nodes is selected in each cube to forward the data packet.

PICS and PRCS protocols [63] use depth information and Time of Arrival (ToA) to calculate the position of the sender and distance between the nodes. PICS uses two hops information to select routing path where as PRCS uses one hop information. Set of nodes is selected using Enhanced Expected Packet Advance (EEPA). EEPA consists of channel quality and path correlation between two neighbors.

2.5 Cluster Based Protocols

Cluster based routing protocols are not suitable for delay sensitive data transmission. The cluster head forwards the packets of all the nodes present in its cluster which causes delay in packet forwarding. Cluster based protocols also become single point of failure if the cluster head node is fixed or there is no alternative path for the cluster nodes in case of cluster head failure.

Cluster-based protocols have low throughput and single point of failure if the cluster head is a fixed node. Distributed Minimum-Cost Clustering Protocol (MCCP) [64] forms clusters based on total energy required by cluster nodes to send data to the cluster head, the residual energy of nodes of a cluster, and the cluster heads relative distance from the sink. Initially all nodes are equal candidates for cluster members and head. Each candidate constructs its set of neighbors to find its own representative clusters based on local information. The average costs of the clusters are exchanged to select one of them as a cluster. MCCP uses a centralized approach to form the non-overlapping clusters. It designates more cluster heads around the sink node to balance the traffic load and cluster heads energy consumption. The Distributed Underwater Clustering Scheme (DUCS) [38] is a cluster based self-organizing protocol. Cluster heads aggregate data received from single-hop cluster nodes and send it to the sink using multihop routing via other cluster heads. Nodes in a cluster keep rotating the role of cluster head among themselves to conserve energy. In the Temporary Cluster Based Routing (TCBR) [65], multiple sinks are used to increase the delivery ratio. Special nodes, called courier nodes, are used to collect data from the normal nodes and transmit to one of the sink nodes in the surface. TCBR is not suitable for time critical applications. Greedy Geographic Forwarding based on Geospatial Division (GGFGD) and Geographic Forwarding based on Geospatial Division (GFGD) [66] are location based 3D routing protocols. A cluster of nodes is considered as a cube. In GGFGD, first the targeted cube (cluster of nodes) is selected and then the next hop is selected from the cube. The targeted cube may be vertex-adjacent, edge-adjacent, or surface-adjacent. The targeted cube is selected if its distance from the sink is shorter than the distance of the current cube, even if it detours the path towards the sink. This may make the total path longer. To avoid this detour in GFGD the targeted cube may only be surface-adjacent. Hence, GGFGD has always longer path then GFGD. But, compared to Multipath Power-control Transmission (MPT) protocol (described in energy efficient protocols) both GGFGD and GFGD have longer paths because they take path loss, transmission delay and residual energy level into account as well. Since GGFGD and GFGD have longer paths compared to MPT, they also have higher average delay.

2.6 Energy Efficient Protocols

The Information-Carrying Routing Protocol (ICRP) [39] is an energy-efficient, real-time, scalable routing protocol. It is not a location-based routing protocol. A source node checks the existing route to the destination when it has packets to send. If there is no existing route then it initiates a route formation process by broadcasting the data packet, which also carries the route discovery message. This way the nodes will keep broadcasting the data packet until it reaches the destination node. All the nodes which broadcast the data packet also maintain the reverse path through which the packet passed. The destination node uses the reverse path information to send the acknowledgment. A new route discovery is required every time the predefined lifetime of the route expires. The Reliable and Energy BAlanced Routing (REBAR) protocol [67] adaptively changes the broadcast domain to balance the energy consumption among the nodes. The nodes near the sink have smaller radius to decrease their chances of being involved in routing packets towards the sink, which increases their working life. The Energy-Efficient Routing Protocol (EUROP) [68] reduces energy consumption by reducing the number of broadcast messages. EUROP nodes use pressure sensors in order to measure their depth, thus avoiding the need for hello messages. This decreases the energy consumption. In Power Efficient Routing (PER) protocol [69], the forwarding node is selected on the basis of distance, angle between the neighboring nodes, and residual energy. A node transmits the packet to the next selected node if the number of received duplicated packets does not exceed the limit. Compared to DBR its energy consumption is lower; however, DBR is not designed to be energy efficient. In the Energy-Efficient Depth-Based Routing (EEDBR) [70] protocol, the nodes share their depth and residual energy information with their neighbors. Each node keeps measuring the residual energy level and if the residual energy level is greater than the threshold value the packet is broadcast. The received packet is held by a node for a certain time based on the residual energy; the nodes with less energy will try to avoid transmission by waiting longer.

The Reliable Energy-efficient Routing Protocol [71] is based on link quality, physical distance and residual energy. R-ERP^{2}R is extension of ERP^{2}R, which took only physical distance and residual energy into account to select the forwarding node. The three metrics are computed and shared by all the nodes. R-ERP^{2}R also avoids delivery of redundant packets by "time-delayed" transmission. The Delay-aware Energy-Efficient routing Protocol (DEEP) [72] is a 3D routing protocol based on energy efficiency and collision rate. There is no handshake mechanism in DEEP. Upper layer nodes forward the data of lower layer nodes for final delivery to the sink. Relay nodes are selected on the basis of link quality and successful delivery ratio. Because of multiple relay nodes, there is a chance that multiple nodes relay the same packet. A sequence number is used to discard duplicate packets. In the Channel Aware Routing Protocol (CARP) [73] the next forwarding node is selected on the basis of hop count and residual energy. Every node knows its hop distance from the sink node. The transmitting node broadcasts a PING packet to find the next forwarding node. If the hop count of the receiving node is less than the transmitting node, then it sends back a PONG packet with information about the residual energy and the link quality. Energy Efficient and Collision Aware (EECA) [74] multipath routing algorithm with multipath power-control transmission (MPT) is based on finding two collision-free routes using constrained and power adjusted flooding. MPT helps to deliver the packet with certain end-to-end packet error rate with minimum transmit power to save the energy.

Mobility Based Protocols: Mobile Delay-tolerant Data Dolphin (DDD) [75] uses mobile nodes (Dolphins) to gather data from the stationary nodes; the speed of the dolphin nodes is a few meters per second. When the dolphin node comes at one hop distance to the stationary node, the latter transmits data to the dolphin node. After getting the data from all the stationary nodes, the dolphin returns to the base station and uploads all the gathered data. Mobicast Routing Protocol for Underwater Sensor Networks [76] is a three dimensional routing protocol based on mobile sinks, i.e. AUVs. Nodes close to an AUV form a 3-D zone known as zone of reference or ZOR. To save power the non-communicating nodes are in sleep mode. The AUVs collect data from the nodes in the ZOR. For nodes to be ready to send data when AUV is close to them, a sleeping node in the next ZOR is woken up in advance by a notification from the AUV.

2.7 Reliable Data Deliver Protocols

Efficient Data Delivery with Packet Cloning (EDDPC) [77] uses the same concept of multiple copies of packet for reliable delivery of data. However, it is a little bit different because it clones, or copies selected data packets when forwarding the data on the basis of link quality and channel conditions. To increase reliability, multiple copies are sent through different paths.

The Resilient Routing Algorithm for Long-term Applications (RRALA) [78] is based on virtual circuit routing. Multihop connections are established in advance between each source and sink and each packet is associated with a particular connection; packets associated with a particular connection follow the same path from the source to the destination. If a connection fails during the packet forwarding then other connection, established in advance, is used. RRALA has not proposed any access method instead it uses modified form of IEEE 802.11 for the simulation.

2.8 Protocols Based on Received Signal Strength

Reliable Energy-Efficient Cross-Layer Routing Protocol RECRP [79] uses RSS to estimate the distance between the nodes and choose the shortest path. ECRP selects the forwarding node based on distance, residual energy of the sender and the potential forwarding node and hop count to the sink node. It is also assumed that the nodes are time synchronized. At the time of data packet routing every data packet contains residual energy and transmission power information. All the neighbor nodes always update their routing table when they receive the data packet even though they are not the relay node. This increases the data packet overhead cost and unnecessary processing for all the neighbor nodes which increases the energy consumption.

NPAN [80] uses the received signal strength to adapt the transmission power to conserve the node energy. The master node initiates the network configuration process by transmitting a Topology Discovery Message (TDM). The nodes will propagate the packet through all the nodes in the network. This way each node will make a list of all its neighbors and forward that list to the gateway to select the most appropriate path for each node. The master node selects path based on available capacity across the path and number of hops. This protocol has some serious problems. First is that, it has centralized path selection process. All the nodes will collect the data and send back to the gateway node which establish routing path for the nodes. If any node goes down, then the master node will not be able to inform the status to the other nodes before the next periodic transmission of TDM. In order to keep the master node updated with changes in the topology more accurately, the periodic transmission of TDM must be quite frequent. [80] used received signal strength (RSS) for short-distance (0-1000 m) measurement between the nodes in underwater acoustic communication. Distance measurement is used to determine the location of the nodes. The Lambert W function also known as product log or omega function, calculates the distance based on received signal strength.

The following protocol cannot be classified in any of the above categories therefore it is described separately. Underwater Wireless Hybrid Sensor Networks (UW-HSN) [81] uses both acoustic wave and radio waves for communication. Radio communication is used for large and continuous traffic above the sea surface and acoustic waves are used for small amount of data underwater. All the nodes are equipped with both kinds of communications. Acoustic waves are used by the nodes when they are underwater and radio waves are used when the nodes are at the surface and communicate with the base station. Nodes are equipped with a piston-based mechanism to move the nodes vertically. When nodes emerge on the surface they send their data to the base station on the ocean surface. All the nodes keep the information of their neighboring nodes using the Turtle Distance Vector (TDV) routing algorithm. TDV is based on Distance Vector Routing (DVR).

2.9 Conclusion

The routing protocols can be classified on the basis of their working principle. We have classified the routing protocols as given below.

1. Location-free protocols: Do not require information of the relative position of the nodes to form choose the routing path.
2. Location-based protocols: Requires information of the relative position of the nodes to form choose the routing path.
3. Opportunistic protocols: Multiple copies of the packet are sent to increase the probability of packet delivery.
4. Cluster based protocols: The deployed nodes are divided into smaller groups. All the nodes in a cluster forward the data packet to a single node known as cluster head and the cluster head forwards the data packet to the sink node.
5. Energy efficient protocols: The focus of the protocol is to conserve as much energy as possible.
6. Reliable data delivery protocols: Reliable data delivery is achieved my multiple copies of the selected data packets or establishing virtual path between the sender and the receiver.
7. Protocols based on received signal strength: Signal strength between the nodes is used to estimate the distance between the nodes and select the nearest node from the shortest routing path.

To substantiate the research work done in this thesis, major shortcomings of the existing protocols need to be highlighted. The summary of the issues is as given below.

1. Some protocols are based on full duplex communication. However, full duplex communication is very difficult to implement in underwater acoustic communication systems because narrow acoustic bandwidth.
2. Most of the routing protocols for UWSN assume that the geographical location of the nodes is known to form the routing paths. However, there are two subtle problems in location-based routing in UWSN. First issue is that the sensor nodes determine their position with respect to the GW, which in turn determines its position via GNSS. Due to varying speed of acoustic waves and the mobility of the nodes, the probability of error in

position determination is quite high. Second issue is that, the position determination by the depth sensors increases the power consumption and cost of the sensor nodes.

3. Some routing protocols like EOAR use timestamp to estimate the propagation delay and some others like PICS and PRCS use timestamp to estimate time of arrival to determine the node position. That requires synchronized time clock between the nodes. Time synchronization in wireless sensor networks is usually achieved by time information exchange between the nodes. However, time synchronization is a challenging task in underwater networks because of variable-propagation delay and node mobility.
4. Requirement of additional devices like depth sensor and array of transducers (for Angle of Arrival method) increase the power consumption of the sensor node. Energy conservation is more important in UWSN compared to WSN because the nodes are very expensive, and their underwater deployment is very costly and difficult.
5. In centralized routing path formation, the sink node forms the routing path for the nodes or cluster heads based on the collected information from the nodes. This approach may produce optimum routes as the sink node has all the information to establish the best routing path. However, changing the routing path in case of a node failure or addition of a node will not be easy. In addition to that, to change the routing path in case of any change in the network requires the nodes to send the information to the central node periodically which decreases the data throughput and increases the energy consumption.
6. As mentioned in section 2.1, the reactive protocols have their own disadvantages. Reactive protocols add packet forwarding delay because of the time required to select the next forwarding node. This delay becomes substantial for time-critical applications in large networks.
7. Cluster based protocols have relatively low throughput. The cluster heads become the bottle neck for the data traffic. Each node sends data to its cluster head which decreases required the transmission power. However, cluster head becomes the point of traffic congestion, particularly for sensor networks that produce high data rate.
8. Cluster based networks and centrally controlled networks pose the threat of single point of failure. When the cluster head is down, then all the cluster nodes get disconnected from the rest of the network until they choose another network node as a cluster head.
9. Opportunistic routing protocols broadcast the data packet and multiple copies of the data packet to the destination node, which increases the probability of data packet delivery. The purpose of sending the multiple copies is to increase the packet delivery reliability. However, this increases the number of transmissions for each forwarding node which may significantly increase the power consumption. Increase in power consumption decreases the nodes operational life as replacement of the nodes batteries is very difficult.

Chapter 3

SOFRP: Self Organized Fast Routing Protocol

3.1 Introduction

As evident from its name, SOFRP is a self-organizing protocol which does not depend on location of the nodes or time-synchronization between the nodes. We have proposed a radial network for SOFRP, as shown in Figure 3-1. Routing is not an easy to solve problem in such networks. The algorithm is designed to recreate the physical topology in the logical routing tables. The paths are established in such a way that only the nodes located in the same radius (called strings, see Section 3.2) will forward the data packets coming from other nodes which are farther from the sink. This way, all the packets are transmitted via the shortest paths. The algorithm uses the knowledge of the topology to initialize the paths. It is designed to avoid collisions and minimizes the overhead due to signaling. All together, these characteristics help save as much energy as possible.

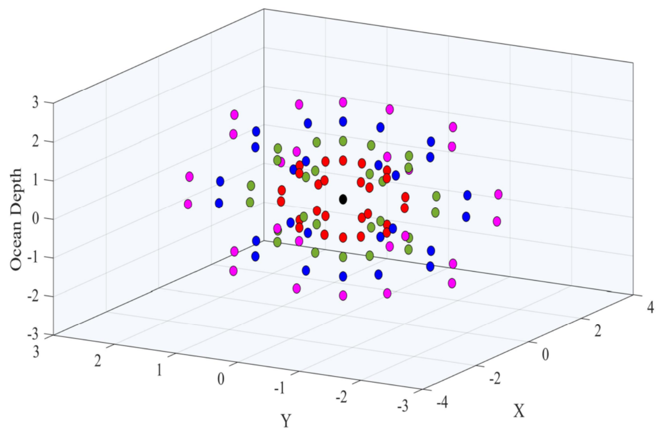


Figure 3-1. A 3D spherical Wireless Sensor Network.

This structure also helps to balance traffic among nodes that are at the same distance from the gateway such that energy depletion is as homogeneous as possible. Other proposals do not consider the physical topology when building the routing trees, and thus they do not balance traffic among routes, causing faster energy depletion of some nodes than others. Obviously, nodes closer to the gateway will forward a larger number of packets, using up energy faster than nodes located farther, which cannot be avoided

Our proposal uses a cross-layer approach by merging some steps of the route creation process with the medium access control layer, in order to minimize the odds of packet collision. The algorithm self-adapts to different network sizes and number of nodes, and, for example, would allow one ship to drop the nodes into the sea and they would find their neighbors and automatically create the paths. It minimizes the cost, in time and computational effort, of forwarding a packet and requires minimal periodic maintenance of the routes.

SOFRP is based on the automatic formation of predefined routing paths to forward the data packet from each node to the sink, without the use of additional measurement devices, such as pressure sensors. However, it is assumed that nodes are placed in a specific formation. The main novelties of this protocol are:

1. It is designed to recreate the physical topology in the logical routing tables, creating the shortest possible paths,
2. The algorithm uses the knowledge of the topology to avoid collisions,
3. It helps to balance traffic among nodes that are the same distance from the gateway,
4. Our proposal self-adapts to different network sizes and number of nodes.

Together, these characteristics help to save as much energy as possible, and minimize the cost, in time and computational effort, of forwarding a packet.

3.2 Network Topology

It is well understood that the larger the number of sensor nodes deployed in an area, the more complete the area monitoring and characterization. Therefore, deploying the large number of sensors in a grid or hexagonal form is preferred for more accurate monitoring. This relationship between nodes density and information accuracy is valid for all kind of monitoring systems, whether terrestrial or underwater, wired or wireless. However, such high accuracy is not always required for all kind of monitoring systems. High density deployment of the nodes has a substantial effect on the cost. Compared to terrestrial monitoring systems, the costs of the underwater sensor nodes and their deployment are very high, especially in deep waters [82]. Therefore, monitoring systems having a low density of sensor nodes are preferred (whenever high accuracy is not required) to reduce the costs. Underwater monitoring systems such as an early warning system for tsunamis [83] and volcanic eruption monitoring [84], [85] will perform well enough with low density sensor networks. In such cases, radial topology, with a larger density at the core, is more suitable to reduce the costs of the monitoring system.

Our protocol is proposed for a kind of hybrid radial/linear topology as shown in Figure 3-2. The reason is that this particular topology is very much suited for monitoring river plumes, where sediments are carried away from a river mouth into the sea, whose scattering follows a triangular pattern, moving sideways with the sea current. It can also be used for monitoring oil spills, coral reefs, and habitats along the coastal area. In general, the topology serves to monitor an area and collect all the information in a single point that connects the network to the outside world.

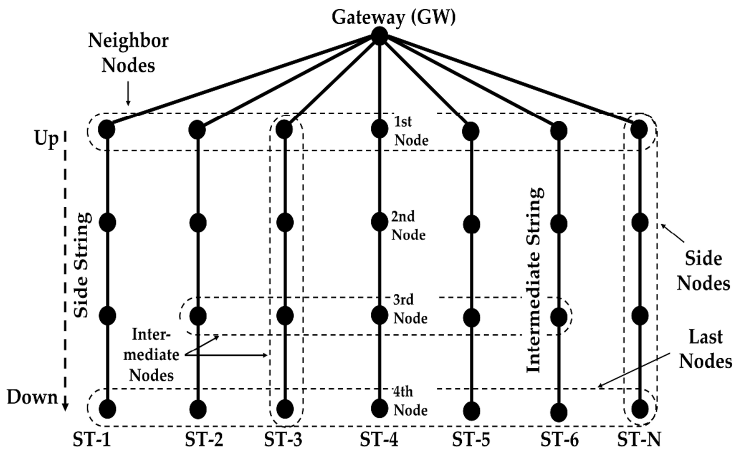


Figure 3-2. Network Topology of N Strings.

We consider that the network contains an indeterminate number of nodes and a gateway (GW), with N radii, where N = {1, 2, …, n}. From now on, we will use the name of string for the radial lines. The strings are shown in vertically in Figure 3-2, but they can be organized as radii departing from the gateway in all directions, distributed over the circle (2D) or the sphere (3D). This topology fits any 3D region shape when only one sink node is possible.

In terms of interference, the worst case is when adjacent strings are parallel to each other, and this will be the case considered. For the purpose of the analysis in this paper, we have considered an equal distance among the nodes in each radius. Varying distances can be considered by setting an appropriate guard time in the transmissions, within a limit. Finally, we consider that nodes adapt their transmission power so that only neighbor nodes can be reached. Thus, nodes that are two or more hops away from the source do not hear its transmissions.

This protocol is designed for an arbitrary number of strings. Along the description, we have mainly considered a network of three strings (Figure 3-3) for easier understanding, but the algorithm is designed for any number of strings.

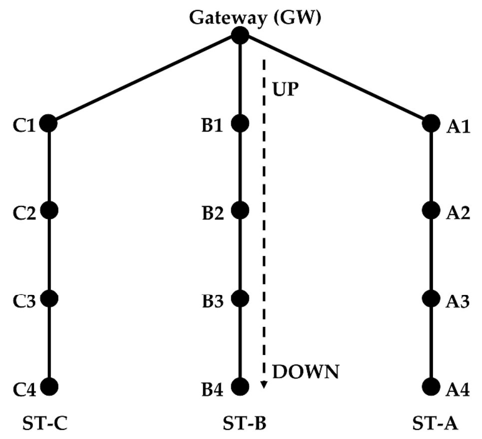


Figure 3-3. Example of network with three Strings.

3.3 SOFRP Overview

The gateway is the node at the surface and relays the data of the other nodes to the terrestrial network. It is also the controlling node of the network. The gateway receives data only from its neighbors. For example, nodes A1, A2, A3, and A4 in Figure 3-3 form a routing path. The nodes in each string forward the data of only those nodes that are above or below them in the same string. When the nodes send data to the gateway, they forward the data to the node above. When the nodes forward control packets from the gateway, they send the packets to the nodes below them.

For explanation purposes, the outer strings are called side strings and all the inner strings are called intermediate strings. Nodes in the outer strings are called side nodes and nodes in the intermediate strings are called intermediate nodes. The nodes that are farther away from the gateway are called down nodes. The transmission range of a node is limited to its adjacent nodes, using power control adaptation according to the distance between nodes.

The routing path formed in this way presents two advantages. First, it creates the shortest path between the sender node and the gateway, which minimizes the delay in terms of the number of hops. If a node were allowed to forward the packet to the adjacent node on the same layer then it would increase the total number of hops, which would increase the delay. Second, it keeps the traffic balance among the gateway neighbor and avoids earlier failure of a gateway neighbor due to excessive transmission.

Initially the nodes have no knowledge about the existence of each other on the network. In order to form the routing path, they need to exchange certain information. The gateway starts the routing configuration process by broadcasting a packet to find its neighbors. The neighbors of the gateway (Figure 3-3) first look for the nodes in their neighborhood that are at the same level (horizontally). This is the set of nodes that is one hop from the gateway, and each of them is the first node in one string (see Section 3.4.1)

It is possible that the packets broadcast by the gateway do not reach some or any of its neighbor nodes due to channel noise. If any of the neighbor nodes fails to receive the broadcast packet, then that node and all the nodes below it will not be able to create the routing path. To address this issue, we have designed the protocol using cross-layer techniques, merging some steps of the route creation process with the medium access control layer. Initially, both the neighbor nodes and the gateway try to discover each other. This is achieved by means of beacon packets, transmitted periodically by each node. When the gateway receives a beacon packet from any of its neighbor nodes it will wait for a specific period and then send an enquiry packet to find its immediate neighbors.

Once the gateway knows the first layer of neighbor nodes, it sends a Route Request (RR) packet to form the strings. The neighbor nodes and those nodes below them forward the request packet, hop by hop, until it reaches the end of each string. Then starts the response process, which sends information about the completion of the string formation. The unique string ID is used by the data packets to send data to the gateway on the corresponding route. When a node receives the data packet, it compares its string ID with the string ID field in the packet header. If both string IDs are the same, then the node forwards the data packet; otherwise, it discards it. Step-by-step detail of the string formation process is given in Section 3.4.2.

To avoid collisions, the nodes transmit the routing packets at different times. A node will randomly select a timeslot (TS) from a set of k timeslots. The length of a timeslot is equal to the sum of the propagation delay, the transmission delay, and a guard time. The duration of k timeslots is called wait period (WP).

Assume that the maximum distance between two nodes is 1000 m. If the maximum packet size is 16 bytes and the channel data rate is 5 kbps, then the time interval is 0.666 + 0.0256 = 0.6916 s (approx.). If we consider a number of 40 timeslots, they would start at instants 0, 0.6916, 1.3832, 2.0748, 2.766, …, 26.972 s. The total of 40 timeslots would last for 27.664s, which would be the WP value. In this configuration, the probability of two nodes selecting the same timeslot out of 40 timeslots is 2.5%. No matter how large the number of timeslots was, the chances of collision are always there.

Different types of transmission modes have been used to optimally achieve delivery reliability and energy efficiency. There are four modes of transmission named mode1, mode2, mode3 and mode4.

1. mode1: the packet is transmitted once without any acknowledgement.
2. mode2: the packet is transmitted three consecutive times without any acknowledgement. The interval between transmissions is equal to one timeslot (TS).

3. mode3: all the candidate nodes send a packet within a period equal to the wait period. Then, acknowledgements are sent during the next WP. Hence, the total time for sending a packet and receiving its acknowledgment is 2*WP. This process of sending a packet and receiving its acknowledgment may repeat two more times in case of packet loss. Therefore, the overall period for the three transmissions remains fixed (which is 6*WP) whether the number of transmissions is one, two or three.
4. mode4: this mode is a little bit different from mode3 in terms of packet retransmissions. In mode4, the packet and its acknowledgement are transmitted three times, even if the packet was successfully delivered in the first or the second transmission.

Different modes of transmission are adopted to avoid unnecessary acknowledgements and interval between the packets transmissions which may increase path formation time and energy consumption of the nodes. Mode1 is used where repeated transmissions and acknowledgement are not required. Mode2 is used where repeated transmissions are required to increase the probability of successful packet delivery, however acknowledgement is not deemed necessary.

Mode3 is a bit more complicated. This mode allows repeated transmission only if previously transmitted packet was not received successfully. This way the energy of the node is conserved. However, the nodes refrain from transmitting the next packet in the sequence of steps to avoid the collision. The wait time WP makes sure that all the nodes in that step of the phase have finished transmitting the packet. One such example is step 5 of Phase 1 (3.4.1).

Mode4 is like mode3, however, the only difference is that the packets are transmitted three times even if the packet was successfully delivered in the first or second transmission. This mode is used where there are chances of collision between the acknowledgement and very high packet delivery reliability is required.

3.4 Routing Path Formation

We can divide the formation of routing paths in two phases:

- Phase 1: Search for Gateway Neighbors
- Phase 2: String Formation

3.4.1 Phase 1: Search for Gateway Neighbors

As we mentioned earlier, in Section 3.3 , the gateway is the controlling node of the network and it can only communicate with its neighbor nodes, which are at one hop distance. The gateway is responsible for starting the process of forming the strings. Since, initially, the gateway does not know about its neighbor nodes, it finds them first. This phase follows these steps:

1. Initially, all the nodes transmit a beacon packet, using mode1, at random time slots to inform of their presence. Other than the gateway, all other nodes ignore the packet.
2. When the gateway receives a beacon packet from any of its neighbor nodes it will wait for a WP. This avoids a possible collision between the SGN packet (see next step) and a beacon packet from another node.
3. The gateway broadcasts the first Search Gateway Neighbor (SGN) packet to find its neighbors (see 1 in Figure 3-4). Altogether, the gateway sends the SGN packet three times using mode2 (Figure 3-5). The SGN packets carry a sequence number.
4. The nodes that receive the first SGN packet wait to receive two more SGN packets before they start finding horizontal neighbor nodes. Suppose that a node fails to receive the first two SGN packets but receives the third SGN packet. This node does not wait for two more SGN packets because it knows from the sequence number that this is the third SGN packet.
5. After receiving the SGN packet(s), each neighbor broadcasts a Search Horizontal Neighbors (SHN) packet to find its horizontal neighbors (see 2 in Figure 3-4) using mode3. Each node selects a random timeslot for transmission.
6. The nodes that received a SGN packet earlier, send an SHN acknowledgment (SHN_ACK) packet. (see 3 in Figure 3-4). The SHN_ACK packets are transmitted using mode3 after all the neighbor nodes have finished sending their SHN packets. The nodes in the lower layer receive the SHN packets as well but they ignore them because they did not previously receive the SGN packet from the gateway.
7. When the waiting time for SHN_ACK packets finishes, each neighbor node sends an SGN response (SGN_RSP) packet to the gateway (see 4 in Figure 3-4), using mode3. The SGN_RSP packet includes information on the neighbors of the sender. When all responses are received, the gateway will know its neighbors and can determine their relative positions by the horizontal neighbor information of each node.
8. The gateway sends back an SGN_RSP acknowledgement (SGNRSP_ACK) packet at a random time, using mode3 as well.
9. To form a string, the gateway assigns a unique string identification (ST_ID) number to each neighbor. The nodes below each gateway neighbor will be part of that straight hop-by-hop forwarding path.

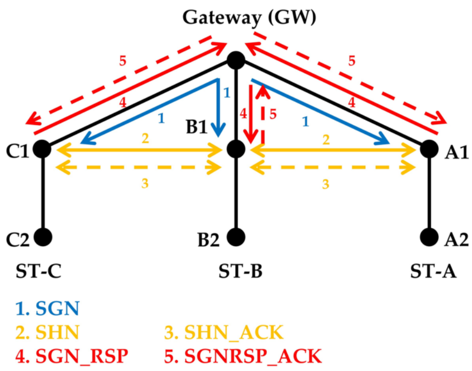


Figure 3-4. Transmission sequence of SGN, SHN, SHN_ACK, SGN_RSP and SGNRSP_ACK between the gateway and the neighbors.

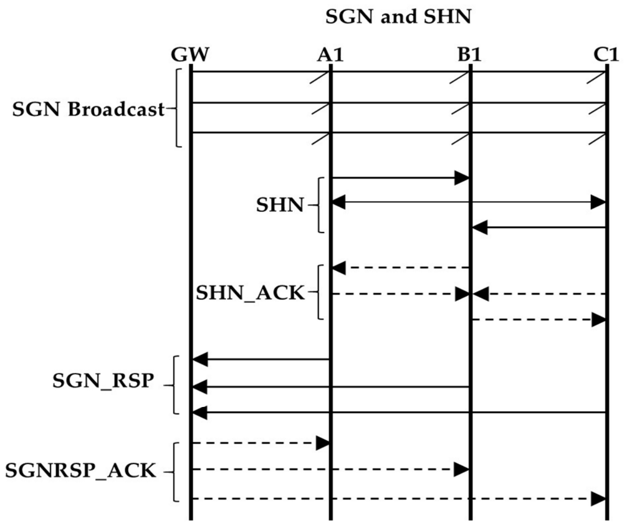


Figure 3-5. The same Transmission Sequence in Message Sequence Diagram.

3.4.2 Phase 2: String Formation

In this phase, the routing paths will be formed. The gateway allocates a string identification (ST_ID) to each neighbor node. The gateway neighbor nodes are called the first nodes in the strings, the next nodes below them are called second nodes, and so on (see Figure 3-2). The process will be completed layer by layer. That is, first, the first nodes in the strings will identify the second nodes in the strings, then the second nodes will identify the third nodes, and this process will continue until the end of each string. In each layer, the identification of the nodes will start from the side nodes and will continue inwards through the intermediate nodes. Step-by-step detail of the procedure is given below:

1. First, Route Request (RR) packets are sent from the gateway to each of its neighbors (see 1 in Figure 3-6) using mode4. The RR packet contains the IDs of all the neighbors and their respective string IDs. When a neighbor receives the RR packet for the first time it checks the list of node IDs to know its assigned string ID.
2. The neighbor nodes will send the RR acknowledgement (RR_ACK) using mode4 as well.
3. After that, first the side nodes in the string will forward the RR packet (see 3 in Figure 3-6), now using mode3. The RR packet will be received by all adjacent nodes within range. These RR packets contain the node IDs with their corresponding string IDs and packet sequence number.
4. When a node receives the RR packet, it records the ID of the sender and its respective ST_ID. The gateway discards any RR packets it receives from its neighbors. The nodes that already have received the RR packet also discard the RR packet.
5. Only the node below in the string will reply with the RR_ACK packet, which will contain the sender ID and the ST_ID. They use mode3.
6. The node that receives the RR_ACK packet as a reply to a previous RR packet transmission records the node and the string IDs in its routing table. This way each node has complete information about the nodes above and below it in the string.
7. Now, the side nodes of the first layer send a Clear to Forward RR (CF_RR) packet (see 5 in Figure 3-6) to their adjacent horizontal nodes (first nodes of the intermediate strings) using mode3.
8. The intermediate nodes send a CF_RR acknowledgement (CFRR_ACK) packet using mode3.
9. The process of sending RR and CF_RR will continue repetitively as described above until all the first layer nodes in all the strings have sent an RR packet to the second layer nodes.
10. When the last intermediate node in the first layer sends the CF_RR packet, it will not receive the CFRR_ACK reply because it is the last intermediate node. It will retransmit the CF_RR packet two more times and then it will decide to send a CFRR_BACK packet.
11. The CFRR_BACK packet will travel from this last intermediate node in the first layer to the side nodes in the second layer (see 9 in Figure 3-6). CFRR_BACK and CFRRBACK_ACK packets will also be transmitted using mode3.
12. When the node of the second layer in the side string receives the CFRR_BACK packet, it starts the process of forwarding the RR packet to the third layer node.

13. Assuming that the number of nodes in all the strings is the same, the last nodes of the side strings will transmit the RR packet three times, but will not get any acknowledgement. Hence, they will conclude that they are the last nodes and will send the CF_RR packet to the intermediate nodes, with information that they are the last nodes. The intermediate nodes also transmit an RR packet, using mode3, and then send a CFRR_BACK packet to the side nodes informing that it is the last node of the string as well. At this stage, all the last nodes randomly select a time slot and send the RR_RSP response (see Figure 3-8).
14. When the RR_RSP packets reach the gateway neighbor nodes, they forward the packet to the gateway. Each neighbor node waits to receive the RR_RSPACK packet from the gateway after transmitting the RR_RSP. After receiving the RR_RSP packet, the gateway knows that the routing path formation is complete for that string. RR_RSP and RRRSP_ACK are sent using mode3.
15. The gateway waits for a threshold period to receive the RR_RSP from all of its nodes. If the RR_RSP packets are not received within that threshold period, the gateway starts a new process of finding the path by sending a new RR packet. A packet sequence number is used so that the nodes know that this RR packet is different from the old one.

3.4.2.1 Purpose of CF_RR and CFRR_BACK Packets

The layered approach to forward the RR packet is to avoid the collision between the RR packets of the adjacent strings. The process of forwarding the RR packet by the neighbors could have been initiated by either the side nodes or the intermediate nodes. In case of intermediate nodes, if the number of strings is odd, then when the CFRR_BACK is sent from the side nodes to the intermediate node, the collision may occur. Therefore, side nodes initiate the process of RR packet forwarding.

The layer by layer approach demands that the RR packet be forwarded by the next layer only when all the nodes of the previous layer have forwarded the RR packet. If any node in the next layer has not received the RR packet from the previous layer, then that node will erroneously form the route with one of its neighbors in the same layer. CF_RR and CFRR_BACK packets are used to ensure that all the nodes in the previous layer have forwarded the RR packet to the next layer and the side nodes in the next layer only start forwarding the RR packet when the previous layer has finished forwarding the RR packet.

Now let us consider two cases of unequal number of the side nodes and intermediate nodes.

1) Case 1: Number of the Side Nodes is Smaller than the Intermediate Nodes
2) Case 2: Number of the Side Nodes is Greater than the Intermediate Nodes

3.4.2.2 Case 1: Number of the Side Nodes is Smaller than the Intermediate Nodes

In case the number of nodes in the side strings is smaller than the number of nodes in the intermediate strings and the side strings have reached the end, the side nodes will send their CF_RR packets to the intermediate nodes before sending the RR response (RR_RSP) packet. When the intermediate nodes receive the CF_RR packet, they will transmit the RR packet and wait for the acknowledgement. Once the intermediate nodes finish the process, they send the Start Response (ST_RSP) packet to the side nodes to inform them that they can initiate the transmission of the RR_RSP packet. In this case, the intermediate nodes send a ST_RSP packet to the side nodes instead of the regular CFRR_BACK packet. The intermediate nodes need to know in which occasions to send the ST_RSP packet. For that, the side nodes send information to the intermediate nodes, in the CF_RR packet, indicating that they are the last nodes and they are waiting for the ST_RSP packet to send the RR_RSP response packet. The ST_RSP and RR_RSP packets are also sent using mode3.

3.4.2.3 Case 2: Number of the Side Nodes is Greater than the Intermediate Nodes

Similarly, if the intermediate strings end before the side strings, the intermediate nodes wait for the ST_RSP packet before sending the RR_RSP packet. When the last intermediate nodes send the CF_RR packet to the side nodes, they also indicate that they are the last node in the string. The side nodes send the ST_RSP response to the intermediate nodes after finishing the process of finding the next nodes. The side nodes also know that the intermediate strings have ended and when they forward the RR packets, they neither send CF_RR to the intermediate node nor wait for the CFRR_BACK from the intermediate nodes. The side nodes simply forward the RR packet to the next node in the string after sending RR_ACK to the node from which they received the RR packet.

If a node was up when the RR packet was forwarded but is down when the RR_RSP packet is sent, then the RR_RSP packet from that particular string will not reach the gateway. In that case, the gateway will send the RR packet again for that string only. In case more than one node is down in multiple strings, the gateway starts the process of route finding for all the nodes again.

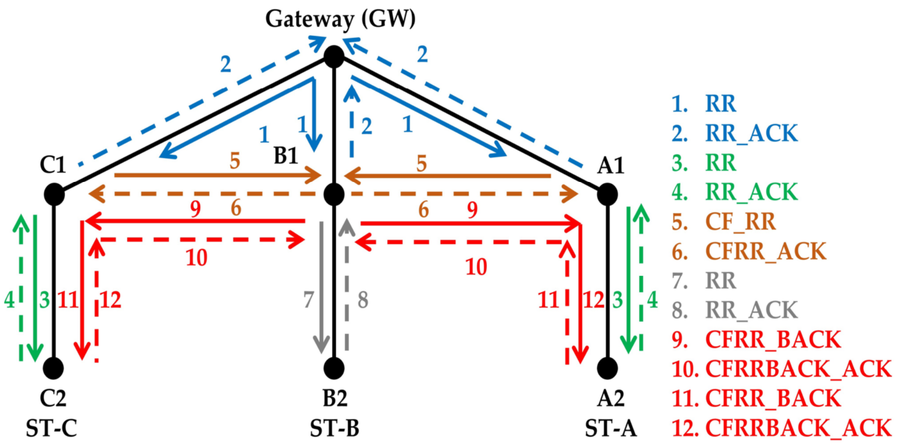


Figure 3-6. Transmission sequence of RR, RR_ACK between the gateway and the neighbors and transmission of RR, RR_ACK, CF_RR, CFRR_ACK, CFRR_BACK and CFRRBACK_ACK between the layers and the adjacent nodes.

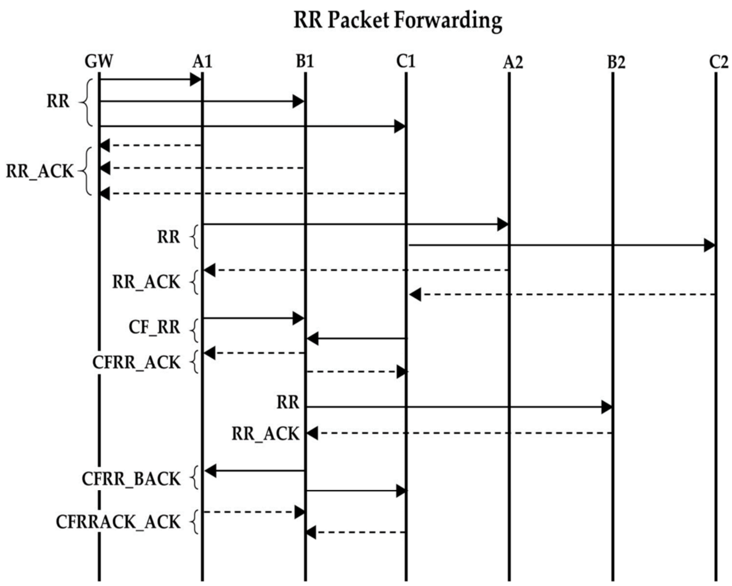


Figure 3-7. The same transmission sequence is also shown in message sequence diagram.

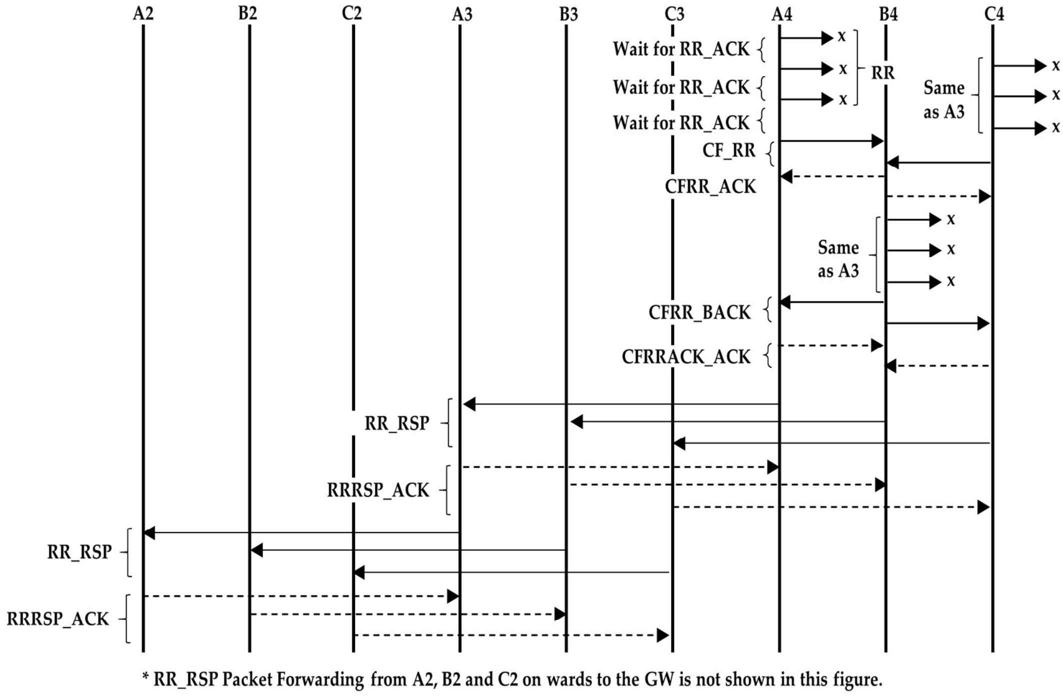


Figure 3-8. Propagation of the RR_RSP message.

3.5 Adaptation for Hexagonal and Grid Topologies

This protocol may also work for a hexagonal topology. When the gateway receives the SGN_RSP packet (see Section 3.4.1) from the neighbors, it can easily deduce that the neighbors are located in a ring around it, because all gateway neighbors indicate that they have two neighbors each. When the gateway sends the RR packet (see Section 3.4.2) to the neighbors, it will set the ring field in the header to true. This will allow a node to form multiple paths with multiple nodes. Each node in a layer may form paths with 1, 2 or 3 nodes in the next layer as shown in Figure 3-9.

The neighbor nodes send the RR packet at randomly selected timeslots. The first RR packet will be received by three next layer nodes and they will form the path with the RR-sending neighbor (shown in Figure 3-9). The rest of the neighbor nodes form two paths with the next layer nodes, except the last node, which will form only one path. Thus, the resulting paths may be unbalanced, but this can be solved either by choosing one of the possible paths or by using all paths in a round-robin fashion, which would balance the energy consumption among the forwarding nodes.

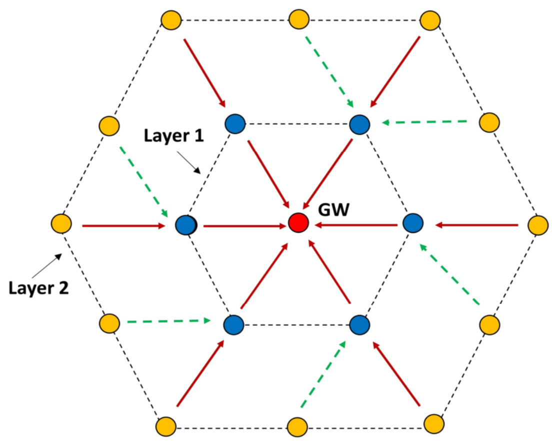


Figure 3-9. Hexagonal Topology.

In the case of a grid topology, the network can be split in several sections as shown in Figure 3-10 and Figure 3-11, each having its own sink (gateway). Each section then can be considered to have the topology shown in Section 3.2 (Figure 3-2 and Figure 3-3).

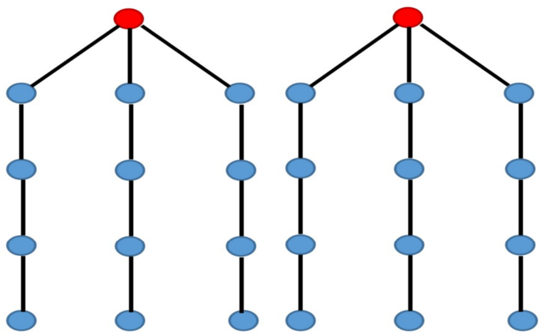


Figure 3-10. 2D Grid.

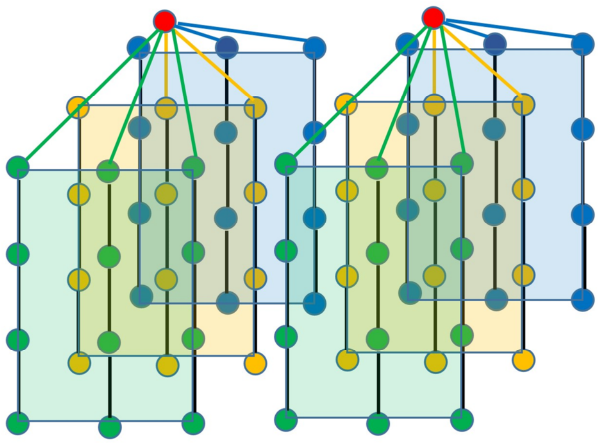


Figure 3-11. 3D Grid.

3.6 String Formation in case of Node Failure

Let us analyze the situation in which one of the nodes is down either at the time of initialization or after the initialization. These two situations and their solutions are discussed below in Scenario 1 and Scenario 2.

3.6.1 Scenario 1: A Node is down at the Time of Routing Initialization.

The down node can be one of the side or intermediate strings. Consider a network with three strings (see Figure 3-12) and assume that C2 is down during the routing initialization. C1 sends the RR packet three times and finally thinks that there are no more nodes in the string. It will send the RR_RSP response as mentioned in Section 3.4.2 paragraph (13). In this case, the RR_RSP is sent not because the string has ended, but because a node failure has occurred. When C1 sends the CF_RR to the intermediate node, it also indicates that it is the last node in the string (although this is not true). Once the intermediate node finishes sending and receiving the RR and RR_ACK packets, respectively, it sends a CFRR_BACK to the side nodes.

Continuing to form a string in this way creates a problem. Suppose that up to C1, ST-C is formed. When B3 forwards the RR packet, it is supposed to receive a RR_ACK from B4 only. But since previously, C3 failed to become part of ST-C, it also responds to B3. Upon receiving two RR_ACK packets, B3 understands that either B4 or C3 is not part of its string. However, at this stage, B3 cannot decide which one.

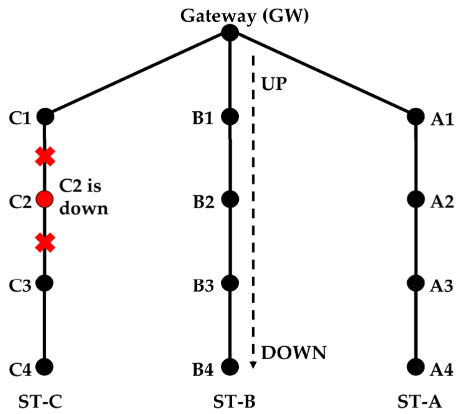


Figure 3-12. A node is down at the time of routing initialization.

Solution

Under this situation, B3 will not send a normal CF_RR packet instead, it sends an information request (INFO_REQ) packet, which asks nodes B4 and C3 (one by one) to send an enquiry packet (ENQ), and waits for the INFO_REQ acknowledgement (INFOREQ_ACK) packet (Figure 3-13). If the INFOREQ_ACK packet is not received after three attempts, then the node waits for a certain time and retries two more times. Assume B3 first asks C3 to send the ENQ packet. When C3 sends the ENQ packet, only the nodes that are already part of a string send an enquiry acknowledgement (ENQ_ACK) packet. Hence, C3 does not receive the ENQ_ACK packet from C4, as it is not part of a string yet (B3 does not send an ENQ_ACK packet because it originally asked C3 to send an ENQ packet). C3 sends an INFOREQ_ACK packet to B3 indicating that it did not receive any acknowledgement. Similarly, B3 asks B4 to send an ENQ packet. B4 will receive the ENQ_ACK packet from A4. B4 provides that information to B3 in the INFREQ_ACK packet. Now, B3 knows that B4 is the intermediate node, because it received the ENQ_ACK packet. B3 sends a cancel (CNCL) packet to C3, which tells it that it has wrongly become part of string ST-B and it should initiate forming a new string. B3 has not sent the CFRR_BACK at this point yet. To form string ST-D, C3 will send a RR packet to C4 and, after receiving the acknowledgement, it will send a CNCL_ACK (acknowledgement) packet to B3. Upon receiving the CNCL_ACK packet, B3 will send a CFRR_BACK packet to C3 and A3 as described before.

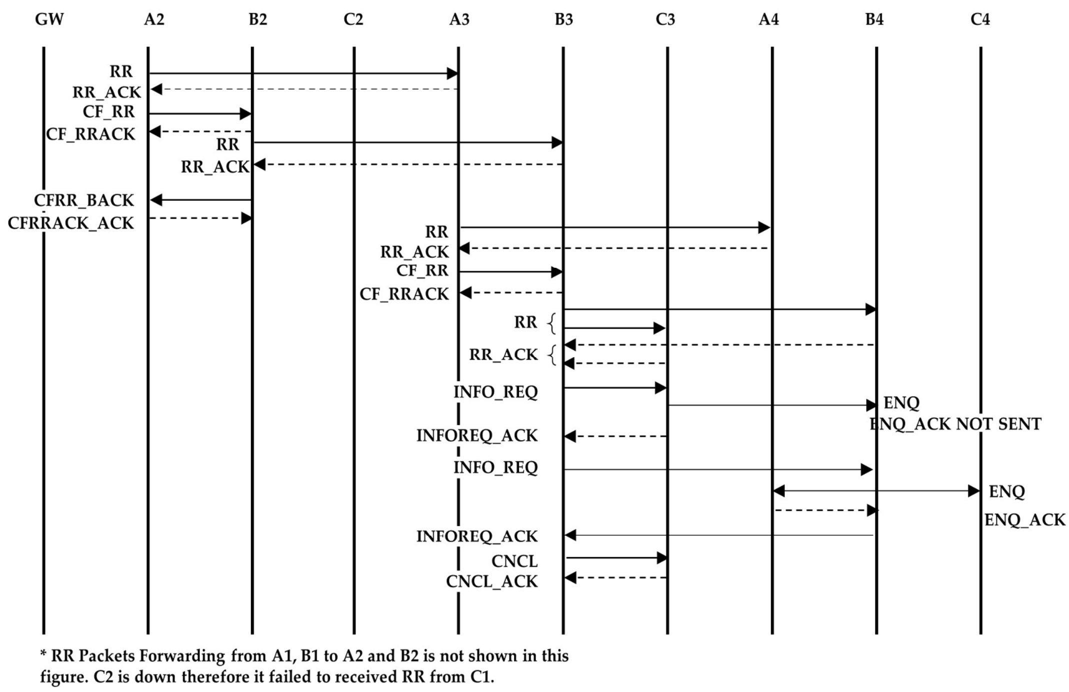


Figure 3-13. Sequence of messages to determine the correct node for string formation.

If the INFOREQ_ACK is not received after three attempts, then the node waits for a certain time and retries 3 more times. Let's assume B3 first asks C3 to send the ENQ packet. When C3 sends the ENQ packet, only the nodes which are already part of a string send an enquiry acknowledgement (ENQ_ACK) packet. Hence C3 does not receive the ENQ_ACK packet from C4 as it is not part of any string yet (B3 does not send an ENQ_ACK packet because it originally asked C3 to send an ENQ packet). C3 sends an INFOREQ_ACK to B3 and tells that it did not receive any acknowledgement. Similarly, B3 asks B4 to send an ENQ packet. B4 will receive the ENQ_ACK packet from A4. B4 provides that information to B3 in the INFREQ_ACK packet. Now B3 knows that B4 is the intermediate node, because it received ENQ_ACK packet from a node. B3 sends a cancel (CNCL) packet to C3 which tells it that it has wrongly become part of string ST-B and should initiate forming a new string. B3 has not sent the CFRR_BACK at this point. To form string ST-D, C3 will send RR packet to C4 and after getting the acknowledgement it will send CNCL_ACK (acknowledgement) to B3. Upon receiving the CNCL_ACK, B3 will send CFRR_BACK to C3 and A3 as described before.

When string ST-D reaches the end node, it starts sending the RR_RSP packet. When the RR_RSP packet reaches C3 it knows that it initiated a new string and the node above is down, therefore it will not forward the RR_RSP packet. In order to send data of ST-D via strings ST-B and ST-C we use the same method of route discovery as mentioned in Scenario 2. This route discovery process starts after the completion of the routing initialization process.

3.6.2 Scenario 2: What will Happen If a Node in a String Goes down after the Initialization?

A mechanism is required to forward the packets in case one of the nodes of a string is down. First, the nodes above and below the failed node in that string should be informed about the failure so they can make an alternative path. Timeout threshold and beacon signals are used to address this issue. If a node does not transmit a packet for a certain period, i.e., 10 WPs, then it should transmit a beacon packet to indicate that it is still alive. The nodes should also expect to receive a data or a beacon packet from the nodes above and below them within the threshold period. If a node fails to transmit data or beacon packets, then the nodes above and below in the string conclude that the node has died and starts the process of finding the alternative path.

Solution

The only solution is that the nodes of the adjacent strings forward the packet. One such alternative path is shown in Figure 3-14. In this example, it is assumed that node C3 has gone down and the data of C4 is forwarded to C2 via B4, B3 and B2.

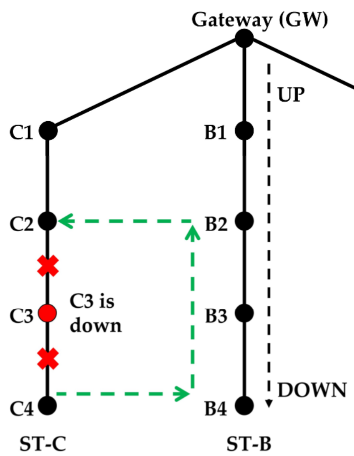


Figure 3-14. Alternative path when a node fails after the routing paths have been formed.

In this way, the traffic load is balanced as much as possible between the nodes of the two strings. If all the packets of all the nodes below the failed node in string ST-C are forwarded via string ST-B only, then nodes of ST-B will be overloaded. A new ST_ID is used for the packets transmitted by C4 to B4. B2 should forward packets of its own string with a different string ID than packets of ST-C. This way, B1 forwards data packets of its own string only in order to create balanced energy consumption. Suppose that B2 forwards a packet of ST-C with a different string ID, say ST-D. The forwarded packet reaches both B1 and C2, but only C2 accepts the packet. When C2 receives packets from B2 with string ID ST-D, it changes the string ID to ST-C before it forwards the packet to C1. Therefore, nodes B2, C2, B4, and C4 handle two different string IDs. When C4 sends or receives data from the node below it (if a node is there), it uses string ID ST-C, but when it sends data to B4 or receives data from B4, it uses string ID ST-D. Two IDs are used by all nodes that form the alternative path.

The node below the failed node initiates the new route-finding process. It broadcasts an RR packet with a new string ID and its existing string ID. Nodes below the initiating node in the same string ignore the RR packet. The nodes in the adjacent string(s) receive the RR packet and recognize that it is sent from the adjacent string. Knowing that, they send back the acknowledgement.

Suppose C4 initiates the new route discovery process. It broadcasts an RR packet, which is received by B4 and it sends back the acknowledgement. Now B4 will forward the RR packet, which will be received by B3 and A4. A mechanism is required such that only the node above (B3 in this case) accepts the forwarded RR packet. For that, B4 adds information in the RR packet, which tells the receiving nodes that this RR packet is not generated by the sending node. When A4 receives the RR packet and it learns that B4 simply forwarded the RR packet of another string, it will not send the acknowledgement to B4. Only B3 will send the acknowledgement to B4. B3 will also forward the RR packet to B2 and it will become part of this new string.

Now, B2 forwards the RR packet, which is received by nodes A2, B1, and C2. Node A2 will ignore the received RR packet because of the different string ID. B1 and C2 both send the acknowledgment to B2. B2 decides to make a path with C2 rather than B1 because it gets a reply from C2, which has the same string ID as C4. B2 sends a packet to B1 to inform that it is not part of the ST-D, and another packet to C2 to inform that it is now part of ST-D as well. This way, a new route is established between nodes C4 → B4 →B3 → B2 → C2.

In case one of the intermediate string nodes fails, the procedure of the new path selection will be almost the same with one exception. Suppose B2 is the failed node and B3 is the initiating node. B3 generates an RR packet, which is received by both A3 and C3. Both A3 and C3 send acknowledgment to B3 along with their energy level information, and B3 chooses the node that has the higher energy level.

3.7 Routing Path Periodic Update Process

The routing initialization process needs to be executed from time to time in order to eliminate any change in the routing path due to movement, failure, or addition of nodes. Before the process is initiated by the gateway, the data communication from the nodes must be stopped, otherwise collisions may occur. For that, the gateway broadcasts an alert packet to all of its neighbor nodes, and throughout the strings. All the nodes that receive the alert send back acknowledgement to the gateway. The gateway already knows about all the nodes in the network. If it fails to receive the acknowledgment of the alert packet from any of the nodes, it sends the alert packet again after waiting for the maximum possible delay. If the gateway still fails to get the acknowledgement, it assumes that the node or nodes are down and starts the routing path update process using the same initial configuration process described above. When a node receives a data packet to be forwarded, it looks into the routing table and compares the string ID of the received data packet with its own string ID. If the string ID of the packet matches with the node's string ID, it forwards the packet.

3.8 Packet Format

A packet is composed of a header and a payload. There are seven header fields, namely S-ID, D-ID, Packet Type, Seq#, ST_ID, Middle Node, and Last Node. The total length of the packet header is 37 bits. The packet header format is shown in Figure 3-15.

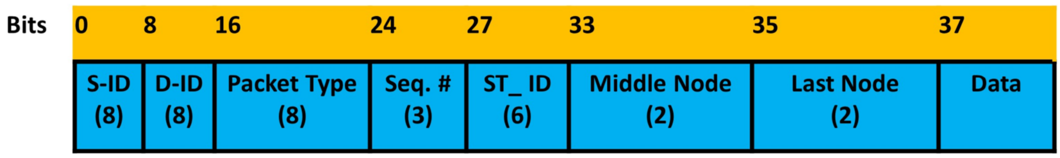


Figure 3-15. Packet Header Format.

The first two fields are the source node ID (S-ID) and the destination node ID (D-ID). The D-ID will be FF for broadcast communication. The packet type field helps the receiving node to identify the type of the packet. Total length of the packet type field is eight bits. Seven bits out of eight bits are used for the unique code for each packet type (shown in Table 3.1) and the last bit (LSB) is used to indicate the ring topology. LSB "1" indicates that the topology is the ring topology. In case of multiple transmissions of the same packet, such as the RR packet, the three-bit sequence number (Seq. #) is used to identify duplicate packets. The string identification (ST_ID) is represented by six bits. With the Middle Node field, the gateway informs its neighbors, in the RR packet, whether they are side nodes or intermediate nodes. The Last Node field is used by the last node in a string when it sends a CF_RR or a CFRR_BACK packet. The list of packet names, their description, and code is given in Table 3.1.

Table 3.1. Packet Descriptions.

Packet Name	Description	Code
BEACON	Beacon	1100011
CF_RR	Clear to Forward the RR	0101000
CFRR_ACK	CF_RR ACKnowledgement	0100001
CFRR_BACK	CF_RR from intermediate node to the side node nodes	1011100
CFRRBACK_ACK	CFRR_BACK ACKnowledgement	1000111
CNCL	CaNCeL	0010001
CNCL_ACK	CaNCeL ACKnowledgement	0010000

Packet Name	Description	Code
ENQ	ENQuiry	0111100
ENQ_ACK	ENQuiry ACKnowledgement	0111100
INFO_REQ	INFOrmation REQuest	0111010
INFOREQ_ACK	INFO_REQ ACKnowledgement	0111010
RR	Route Request	0000111
RR_ACK	Route Request ACKnowledgement	0100101
RR_RSP	RR ReSPonse	0101101
RRRSP_ACK	RR_ReSPonse ACKnowledgement	0001101
SGN	Search Gateway Neighbor	0101010
SGN_RSP	SGN ReSPonse	0011001
SGNRSP_ACK	SGN_ReSPonse ACKnowledgement	0001100
SHN	Search Horizontal Neighbor	0001011
SHN_ACK	SHN ACKnowledgment	0100000
ST_RSP	Start sending the RR ReSPonse	0110001
STRSP_ACK	ST_RSP ACKnowledgment	0110101

3.9 Performance Analysis

We have developed a mathematical model to calculate the convergence delay, that is, how long it would take to create the routing paths and compared it with the simulated convergence delay. The calculations are carried out with the help of MAPLE®. The list of the symbols used in the following equations is given in Table 3.2.

Table 3.2. List of Equation Symbols.

Symbol	Description
DCFRR	Delay for CF_RR and CFRR_ACK packets delivery
DCFRRBACK	Delay for CFRR_BACK and CFRRBACK_ACK packets delivery
DGWRR	Delay for RR packet delivery to the gateway
DRR	Delay for RR packet delivery start at first layer
DRRALL	Total delay for RR packet to be delivered to all nodes
DRRRSP	Delay for RR_RSP and RRRSP_ACK packets delivery
DSGN	Delay for SGN packet delivery
DSHN	Delay for SHN and SHN_ACK packets delivery
DSHNRSP	Delay for SHNRSP and SHNRSP_ACK packets delivery
DTOTAL	Total delay for RR_RSP packet to reach the gateway
DR	Data rate (bits/sec.)
DT	Distance between two nodes (m)
GT	Guard time (sec.)
LTS	Duration of a wait period (sec.)
ND	Number of nodes in a string

Symbol	Description
PKD	Packet transmission delay (sec.)
PROD	Propagation delay (sec.)
PS	Packet size (bits)
RPH	Repetition of RR packet forwarding within a layer
RPV	Repetition of RR packet forwarding through all layers
SPS	Sound Propagation Speed (m/s)
ST	Number of strings
TS	Duration of a timeslot (sec.)
TTS	Number of timeslots in a wait period
WP	Wait Period

For the mathematical analysis, we will assume that the number of nodes in all the strings is the same. First, we will calculate the duration of one timeslot. This is the amount of time reserved for a packet to reach from one node to another. This value depends on two factors: the packet size and the propagation delay. Equation 3.1 gives the size of the packets in bits. Equations 3.2 and 3.3 calculate the packet delay and the propagation delay respectively. Equation 3.4 calculates the guard time to take into account variations in propagation delay. The guard time is 5% of the total of the packet transmission delay and the propagation delay. Equation 3.5 adds packet transmission delay, propagation delay and guard time to obtain the duration for one timeslot.

$$PS = 37 + 8 * ND, \quad 3.1$$

$$PKD = PS/DR, \quad 3.2$$

$$PROD = DT/SPS, \quad 3.3$$

$$GT = (PKD + PROD) * 0.05, \quad 3.4$$

$$TS = PKD + PROD + GT, \quad 3.5$$

Next, we developed the equations to calculate how long the gateway takes to find its neighbor nodes plus the time it takes the neighbor nodes to find their horizontal neighbors. Equation **¡Error! No se encuentra el origen de la referencia.** calculates the length of the wait period. As mentioned in Section 3.4.1, the gateway waits one WP before transmitting SGN packets in order to avoid a collision with any of its neighbors, which might be broadcasting a beacon packet (see Equation **¡Error! No se encuentra el origen de la referencia.**). Then the gateway sends an SGN packet three times. Equations **¡Error! No se encuentra el origen de la referencia.** and **¡Error! No se encuentra el origen de la referencia.** give the delay for the SHN and SHN_ACK packets and for the SGNRSP and SGNRSP_ACK packets, respectively.

$$WP = TS * (TTS - 1), \quad 3.6$$

$$DSGN = WP + 3 * TS, \quad 3.7$$

$$DSHN = 6 * WP, \tag{3.8}$$

$$D_{SGNRSP} = 6 * WP \tag{3.9}$$

The delay for the pairs of SHN and SHN_ACK packets, RR and RR_ACK packets from the gateway to the neighbor nodes and RR_RSP/RRRSP_ACK packets is shown in Equation 3.10. Two WPs are required for packet sending and receiving and, since the packets will be sent three times, total WPs will be 2 * 3 = 6.

$$D1 = 6 * WP \tag{3.10}$$

The delay for the pairs of RR and RR_ACK, CF_RR and CFRR_ACK and CFRR_BACK and CFRRBACK_ACK packets is shown in Equation 3.11:

$$D2 = 3 * D1 \tag{3.11}$$

The process to send the RR and CFRR packets proceeds from both sides of the network to the intermediate node. The number of intermediate nodes is even if the number of strings is even, otherwise there will be an odd number of intermediate nodes. Merging both scenarios, the process is repeated half times the number of strings and the result is rounded up to the next integer for an odd number of strings (Equation 3.12). As mentioned in Section 3.4.2 the RR forwarding process is completed layer by layer and this process is repeated (ND-1) times (Equation 3.13).

$$RPH = \text{ceiling}\ (ST/2), \tag{3.12}$$

$$RPV = ND - 1, \tag{3.13}$$

If we add all the delays calculated up to this point, we obtain the total delay for RR packet forwarding from the gateway to all the nodes in the network (see Equation 3.14). After the RR packet has been forwarded, the last nodes in the strings start sending the response. Since all the nodes of each layer send an RR_RSP packet at randomly selected timeslots, we simply multiply the delay for one pair of RR_RSP/RRRSP_ACK packets (Equation 3.15) by the number of nodes in the strings. Finally, to get the total delay, we add Equations 3.14 and 3.15 to get Equation 3.16. Equation 3.17 is the expansion of Equation 3.173.16.

$$D_{RR\_ALL} = RPV * (RPH * (D2)) + D1, \tag{3.14}$$

$$D_{RR\_RSP\_ALL} = ND * D1, \tag{3.15}$$

$$D_{TOTAL} = D_{RR\_ALL} + D_{RR\_RSP\_ALL}, \tag{3.16}$$

$$D_{TOTAL} = \text{ceiling}\ (ST/2) \ \ * ((ND-1) * (D2)) + D1 + ND * D1 \tag{3.17}$$

This mathematical analysis has been validated via simulation, as described in the next section. Several cases have been simulated to assess the convergence delay, such as when the number of strings and nodes varies (see Figure 3-16, Figure 3-17 and Table 3.4,

Table 3.5), and for a varying number of timeslots in the wait period (Figure 3-22). A comparison between the values obtained via the mathematical equation, Equation 3.17, and via the simulations is shown in Figure 3-22. As it can be seen, there is an almost perfect agreement between both methods.

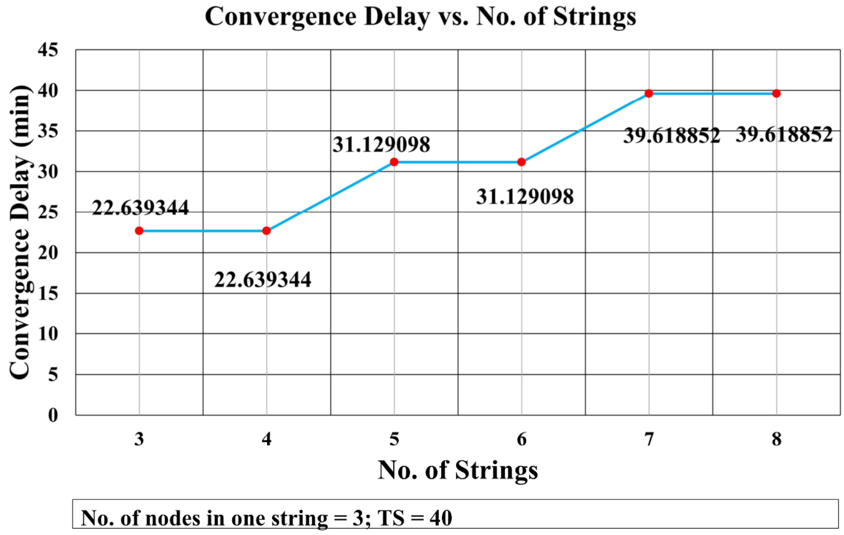


Figure 3-16. Convergence Delay vs. No. of Strings.

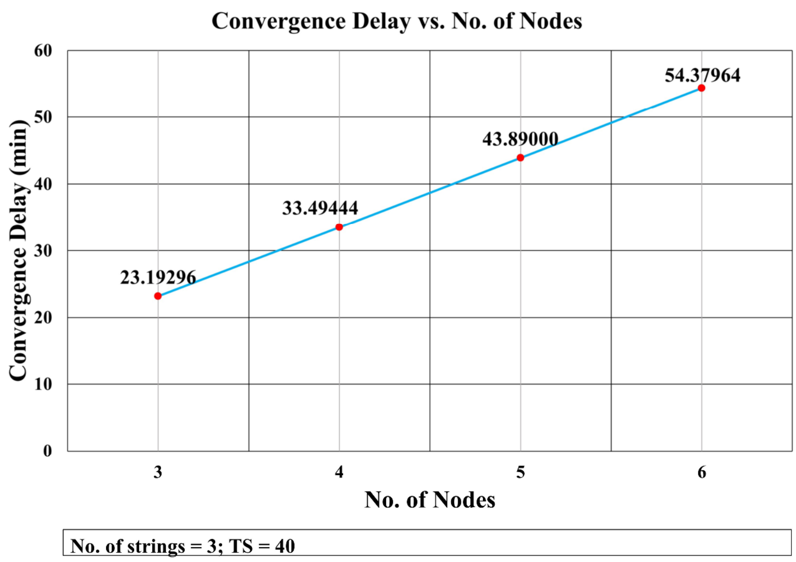


Figure 3-17. Convergence Delay vs. No. of Nodes.

3.10 Computer Simulation Results

The performance of the protocol is analyzed in terms of three parameters, i.e., convergence delay, probability of collision between packets and number of retransmitted packets in case of packet loss. Convergence delay is defined as the total time required to form the routing paths, which starts with the transmission of the beacon packets and ends when the gateway successfully receives the RR_RSP packets.

For the simulations, we have assumed a topology with the structure shown in Figure 3-3. The distance between the gateway and its neighbor nodes is set at 500 m. Similarly, the distance between two adjacent nodes in a string is also set at 500 m. The number of nodes in all the strings is the same. Propagation speed is taken as 1500 m/s. The data rate is 5000 bits per second and the packet size is calculated using Equation 3.1. The transmission power considered for the simulations is 18 W (a value taken from Reference [86]), and the BER is 10−6. The simulation parameters are summarized in Table 3.3. All these values are realistic in UW communication networks. The simulation tool is a proprietary MATLAB®-based simulator.

Table 3.3. Simulation Parameters.

Parameter	Value	Unit
Sound speed	1500	m/s
Distance between nodes	500	m
Data rate	5000	bits/s
Header size	37	bits
Transmission power	18	Watts
Bit error rate	10–6	
Number of runs	500	

First, we analyze the effect of increasing the number of strings on the convergence delay. Figure 3-16 shows the results of a simulation in which there are three nodes in each string, but the number of strings varies. It shows that the convergence delay remains the same when the number of strings increases from odd to even, as for example from three to four or from five to six. This is due to the fact that the number of transmissions of RR packets and CF_RR packets remains the same when the number of strings increases from odd to the next even number. Hence, the delay remains the same for a number of strings (1, 2), (3, 4), (5, 6), and so on. The difference of delay between string 4 and 5 is 8.489754 min. Similarly, the difference between strings 6 and 7 is also 8.489754 min.

The effect of increasing the number of nodes in a string is depicted in Figure 3-17. It shows that the convergence delay increased linearly with the number of nodes. This was as expected because the delay incurred by each layer is almost constant. The delay increased as the number of nodes grew due to the increase in the packet size. The difference in the packet delay transmission was constant (0.0016 s) for consecutive increases in the number of nodes (see Table 3.4). The impact of augmenting the network in one node per string in the total convergence delay was 0.091728 (see

Table 3.5).

Table 3.4. Packet Delay Analysis

Nodes (ND)	Packet Delay (PKD)	$PKD_i - PKD_{i-1}$
3	0.0122	
4	0.0138	0.0016
5	0.0154	0.0016
6	0.0170	0.0016
7	0.0186	0.0016

Table 3.5. Convergence Delay Analysis

Nodes (ND)	Convergence Delay (CD)	$\alpha = CD_i - CD_{i-1}$	$\beta = \alpha_i - \alpha_{i-1}$
3	22.639344		
4	32.694753	10.055409	
5	42.841890	10.147137	0.091728
6	53.080755	10.238865	0.091728
7	63.411348	10.330593	0.091728

We have analyzed the number of packet retransmissions when the probability of packet loss increases from 0 to 20% at intervals of 2% (Figure 3-18). The graph shows that, even at high packet loss probabilities, the number of retransmitted packets was in the range of 11–13%.

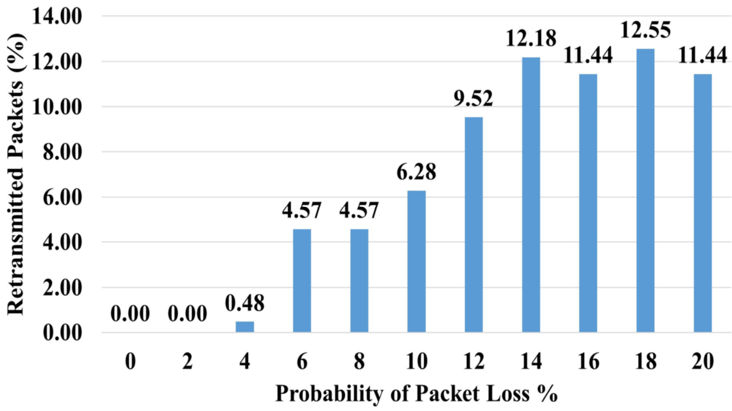


Figure 3-18. Number of Retransmitted Packets vs. Probability of Packet Loss.

The collision rate is also tested. Figure 3-19 shows the collision rate when the number of nodes is constant (three nodes in each string), whereas the number of strings increases from three to eight. The graph shows that the collision rate increases nonlinearly with the number of strings. It shows that there is a slight change in the collision rate when the increase in the number of strings is from an odd number to an even number. From even number of strings to the odd number of strings, the change is negligible.

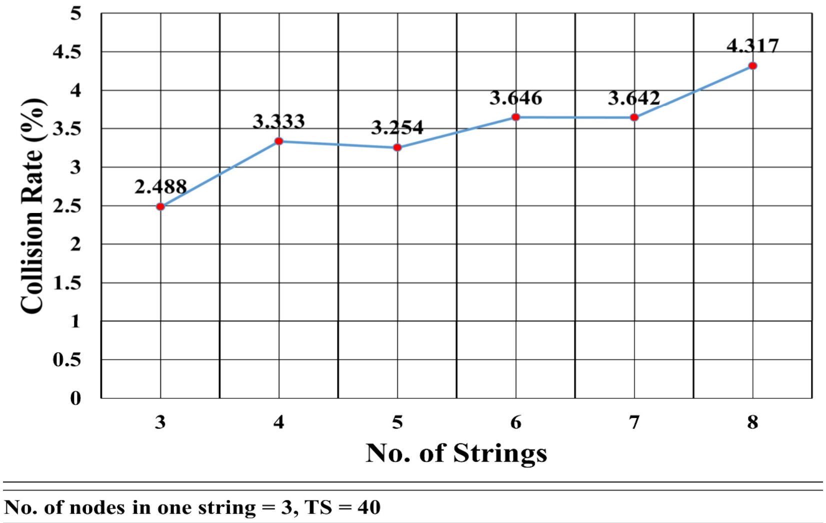
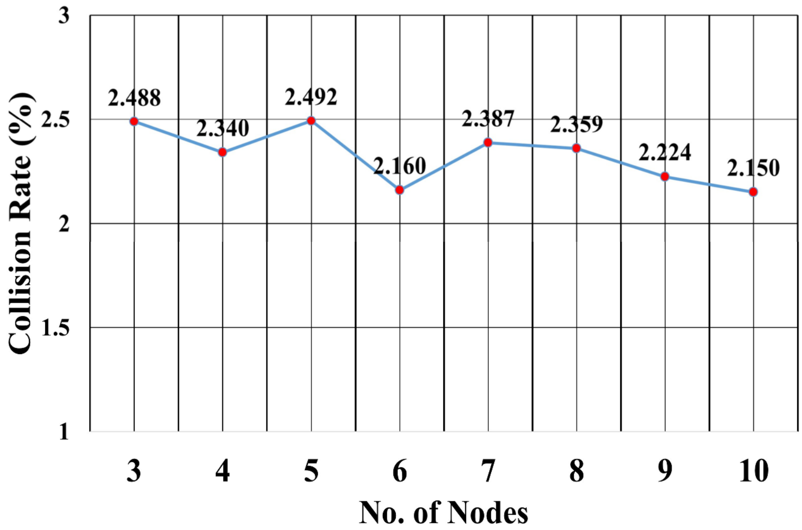


Figure 3-19. Percentage of Collision Rate vs. No. of Strings.

Figure 3-20 shows the collision rate as the number of nodes per string increases from 3 to 10. There is no clear relationship between the number of nodes in a string and the collision rate. It seems that the collision rate remains roughly constant. The mean is $\mu = 2.325$ and the standard deviation was $\sigma = 0.135$. This was due to the fact the collision among the nodes mainly depends on the number of strings rather than the number of nodes in a string.



No. of strings = 3, TS = 40

Figure 3-20. Collision Rate vs. Number of Nodes.

The number of timeslots is an important factor in controlling the collisions between transmitted packets. Figure 3-21 shows the simulation results for different numbers of timeslots ranging from 10 to 80. This comparison shows the tradeoff between collision rate and convergence delay. It is clear from the graph that increasing the number of timeslots decreased the collision rate, but it increased the convergence delay. When using 10 timeslots, the convergence delay is minimum, but the collision rate is maximum, whereas in case of 80 timeslots, the number of collisions was at a minimum, but the convergence delay was at a maximum. From Figure 3-21, it is observed that the optimum number of timeslots is around 30. We have chosen 40 timeslots in order to have a low collision rate, though at the cost of a little higher convergence delay.

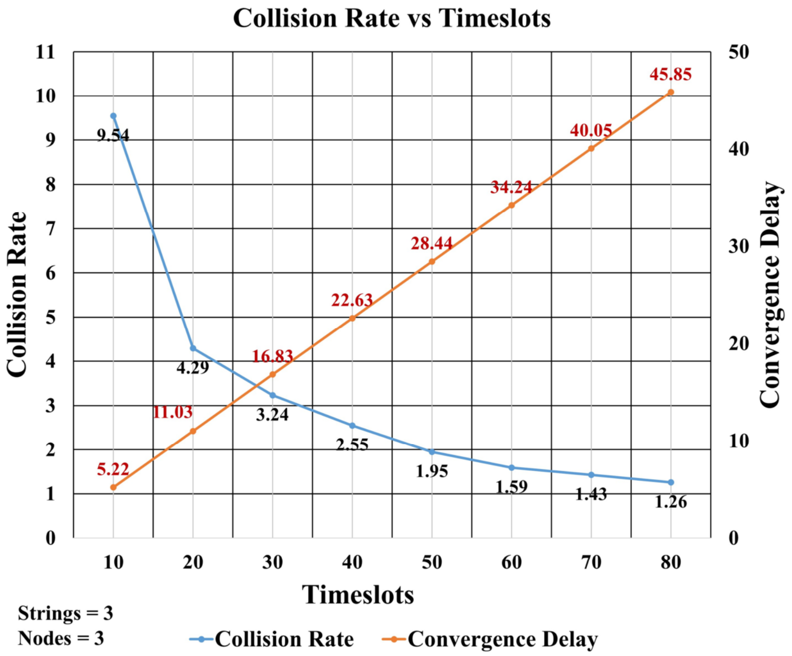


Figure 3-21. Collision Rate vs. Convergence Delay.

Figure 3-22 compares the convergence delay obtained from the mathematical equations and from the simulations (see green bars). It can be seen that there was almost perfect agreement between these values. Figure 3-22 also includes a comparison between the simulated collision rate values and the calculated collision rate. It also shows that the simulated collision rate was in agreement with the theoretical analysis.

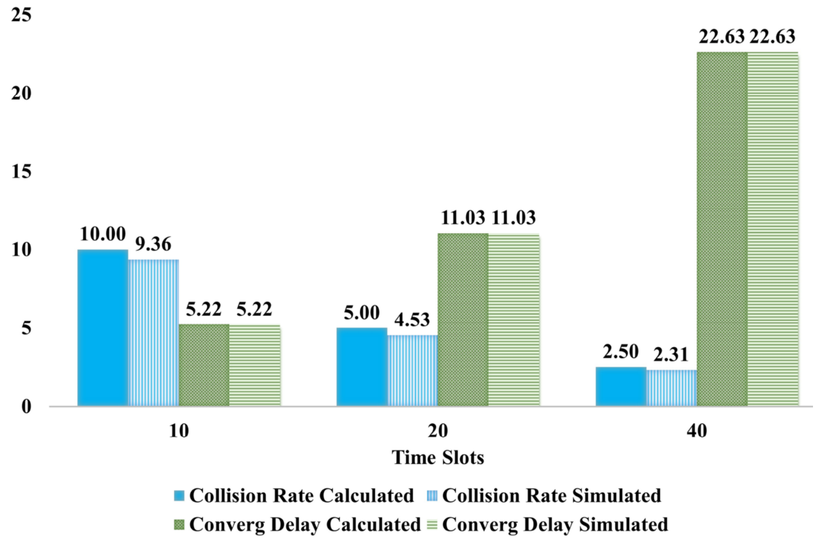


Figure 3-22. Comparison of Time Slots for Collision Rate and Convergence Delay.

It can be seen in **Figure 3-23** that the average energy consumption for 10 timeslots was higher than for 20 or 40 timeslots. This happens because the number of collisions in case of 40 timeslots was lower than in the other scenarios and, as a result, the number of retransmitted packets was lower.

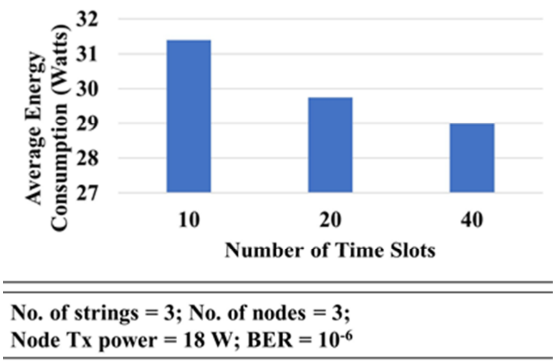


Figure 3-23. Energy Consumption vs. Number of Time Slots.

3.11 Discussion of Results

If we analyze the results, we can see that the convergence delay was due to the long propagation delay of the underwater acoustic channel. In order to avoid collisions, some kind of time synchronization must be used in this protocol. For example, it is possible that RR packets sent by the first nodes of the side strings have been successfully received by the second nodes, but we must wait for a period equal to three transmissions of RR and RR_ACK packets before the side nodes transmit the CF_RR packet to the intermediate nodes. The reason is that the two side nodes do not know about each other transmission's state. If one side node has successfully sent the RR packet but the other one needs one more transmission, then synchronization will be lost, which may cause collisions at later stages.

Figure 3-17 shows that the convergence delay increases linearly with the number of nodes in the strings. In Figure 3-16, we see that the convergence delay changed only when the number of strings increased from even to odd. Figure 3-19 shows that the collision rate also increased substantially when the number of strings grew from odd to even, whereas there is a trivial increase in the collision rate when the number of strings grows from even to odd. However, Figure 3-19 shows that, when the number of nodes increased, the collision rate remained almost the same with an average of 2.325% and standard deviation of $\sigma = 0.135$.

The percentage of retransmitted packets increased with the probability of packet loss, but it was not high (see Figure 3-18). Under normal channel conditions, we may expect low packet loss probability and hence a small number of packet retransmissions.

We can compare our proposal with the location-free protocols described in Section 2.2. Both NLPU [40] and SOFRP are proactive protocols, using a probe message to establish the path and their topology has one gateway node. However, there are two main differences: a) NLPU does not replicate the physical topology, thus paths may be longer than required; and b) the channel is split into several sub-channels, thus decreasing the available bandwidth per node.

Assume a network that has 30 nodes, organized in 5 strings and 6 rows. The distance between the nodes is 1000 m and data rate is 10 kbps. The communication channel is divided into seven subchannels, each with a data rate of 1428 bps. If we analyze the end-to-end delay achieved with NLPU and SOFRP when the farthest node transmits, we find that for NLPU, this value is 20.3 s, while SOFRP achieves an end-to-end delay of 4.3 s. Overall, an improvement of roughly 75%.

It is not possible to carry out a proper comparison with the other location free protocol, LFLSR, as the results shown in Reference [42] do not describe the network configuration and information on the number of hops each packet requires is missing. However, the use of periodic Hello packets imposes an additional burden that increases the end-to-end delay of packets. Hence, LFLSR would at least require 0.667 s more than SOFRP per each Hello packet transmission that the data packet encountered along the way.

3.12 Conclusion

In this chapter, we have presented a routing protocol adapted for acoustic underwater networks with a hybrid radial/linear topology. It can also be easily used in hexagonal or rectangular grids by just allowing the formation of multiple connections via one intermediate node. Such networks are an interesting topology for monitoring medium-size oceanic regions such as estuaries, fishing zones, or geological features of interest. The protocol is designed to minimize the delay for sensitive data transmitted by the nodes, minimize the costs of managing such a network, and to provide strong resilience.

Retransmission of the packets because of the collision at the receiver causes increase in power consumption of the sensor nodes. We have reduced the energy consumption by designing the protocol in such a way that the packet collisions do not occur during the initialization and maintenance of the routes. Furthermore, the selection of the forwarding node is not based on the depth information of the node.

As a proactive protocol, this proposal achieves a low packet forwarding latency compared to reactive protocols. Although the convergence time at initialization is high due to the effort made in the design to avoid collisions, once the routing paths have been formed, the packet forwarding delay is quite low, thus reducing computational cost.

The gateway initiates the path formation process, assigns a unique identification for each string, and serves as a connecting point with external networks. The paths are created in a hybrid radial/linear way, by forwarding a Route Request packet that travels along the network and, upon reaching the end of each string, a response is sent back. Cross-layer techniques have been used to minimize the odds that some node is mistakenly considered down during path initialization.

To counteract node failures, the protocol includes mechanisms for creating alternative paths, thus making the protocol fault tolerant. The full set of possible scenarios has been analyzed in the paper. The protocol provides viable solutions for all of them without compromising the transmissions of the other nodes.

A performance analysis has also been presented, and it has been validated through simulations. The analysis of results shows that extending the network, both in number of nodes per string and in number of strings, has a linear impact on the performance.

Chapter 4
SPRINT: Self-organized Proactive Routing Protocol for Non-uniformly Deployed Underwater Networks

4.1 Introduction

Underwater wireless sensor networks (UWSNs) have many applications related to environmental monitoring, disaster alerts, and military surveillance. UWSN may be deployed in either shallow or deep waters. When a sensor network is deployed in shallow waters and sensors are attached to the bottom of the sea, a 2D deployment of the network is considered. In deep waters, the nodes are suspended by a rope or a chain, which is attached to the surface buoys or the anchors at the bottom of the sea (Figure 4-1). The suspended sensor nodes in 3D networks keep moving in all directions. This movement is limited by the length of the connecting rope or chain. The continuous random movement of the nodes makes the protocols design more challenging in 3D networks.

In a cooperative network, the nodes farther away from the sink node, also known as the gateway (GW) node, send their data to the GW with the help of intermediate nodes known as relay nodes. To select a relay node, the position of its candidate nodes should be known. For example, nodes A, B, C, D, E, F, and G in Figure 4-2 are at one hop distance from node S. All of these nodes will receive the data transmitted by node S, but only the nodes closer to the GW should retransmit the data. Therefore, before node S can select one of these nodes, it needs to know the position of nodes A, B, C, D, E, F, and G.

In WSNs, determining the location of a sensor node is trivial. The sensor nodes can directly determine their position with the help of Global Navigation Satellite Systems (GNSS). However, this process in UWSNs is a challenging task because the GNSS signal is not present. One way to determine the position of the underwater nodes is that the GW gets its position with help of GNSS and the rest of the nodes determine their position with respect to the GW. However, in a 3D network, it is very difficult to determine their position by this method because of the continuous limited movement of the nodes.

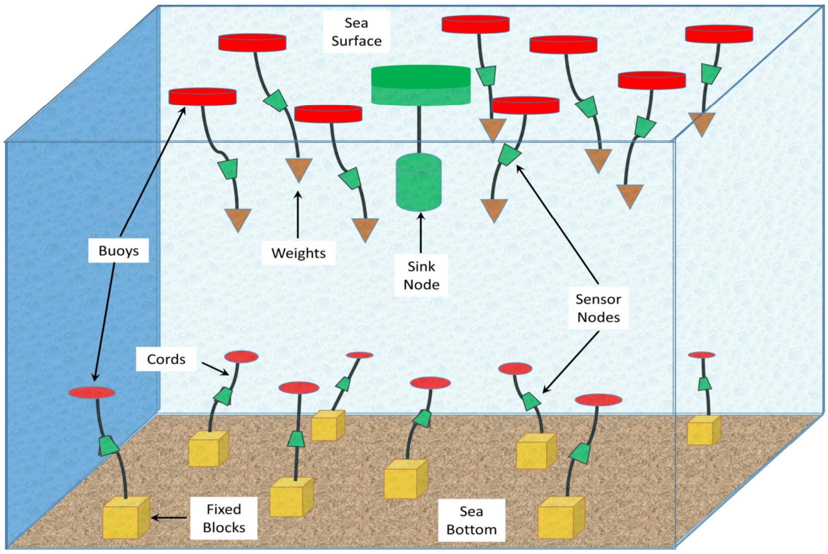


Figure 4-1. 3D UWSN Architecture.

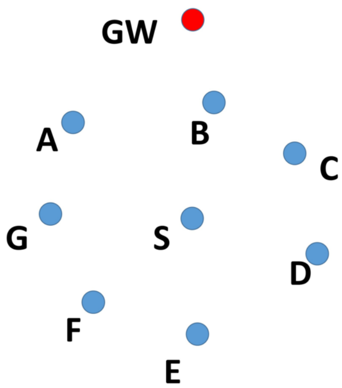


Figure 4-2. Selection of Relay Nodes.

In UWSNs, the position of the nodes can also be determined by Time Difference of Arrival (TDoA) [87] and Time of Arrival (ToA) [87]. TDoA can be applied in two ways: 1) Multilateration 2) Transmission medium difference. In the first method, two or more sensor nodes, at different locations receive a beacon packet from a sensor node. The location of the beacon receiving nodes is known and the time of the transmission is provided in the beacon packet. The difference in the arrival time of the signal due to different positions of the receivers is used to calculate the location of the beacon sending node. In the second method, two different transmission-media, like radio frequency and acoustic wave are used. The distance is estimated by different arrival times due to the dissimilar velocities of the radio wave and acoustic wave [88], [89]. However, EM waves are not available and TDoA cannot be used in underwater environments. The time of Arrival (ToA) technique is an alternative approach; it is based on the travel time of the acoustic wave from the source to the destination. The sender stamps the time of sending the packet and the receiver calculates the travel time by comparison to its local time and estimates the distance. This method requires time synchronization between the sender and the receiver nodes. Achieving time synchronization among the nodes in a UWSN is very difficult because of variance in the acoustic waves' propagation speed and limited mobility of the sensor nodes caused by the water currents.

As an alternative method, the position of the nodes can be determined by measuring their distance from the water surface by means of depth sensors. This method is used by many routing protocols, like DBR [55], LB-AGR [47], and VBF [90]. However, depth sensors have their own disadvantages, like measurement errors, increase in power consumption due to depth sensing and computation, and the extra cost [91].

Because of the constraints and disadvantages mentioned above, the routing protocols which are based on neither the location of the sensor nodes or use the depth sensors are more practical and beneficial. In this paper, we use the distance between the sender node and the candidate nodes as a metric of selection for the relay node, estimated by the received signal strength (RSS).

After a review of the state of the art of underwater routing protocols in Chapter 2 , the three metrics used in the new protocol proposed are explained in detail in Section 4.2. Later on, the process to build the routes is examined (Section 4.3) and a mathematical analysis is performed (Section 4.5). Next, a suitable packet format is established in Section 4.6, and results of simulations are presented (Section 4.7) with a discussion of their importance. Finally, the final conclusions are remarked.

4.2 Overview of the SPRINT Protocol

The RSS metric is used to estimate which node is the closest neighbor of the transmitting node. The node selects the node that is closest to it to forward the packet. Along with the distance, two more metrics are used: number of hops between the sender and the sink, and the number of neighbors of the candidate nodes. The minimum distance between the nodes also increases the probability of successful packet delivery.

The estimation of the RSS is not very precise due to the path loss, the fading, and the limited mobility of the sensor nodes. The spreading of acoustic signal is an important factor for path loss. Spreading losses depend not only on the transmission range, but also on the propagation model adopted: cylindrical (shallow waters) or spherical (deep waters). The path loss is also due to absorption. The absorption of acoustic channels depends on the frequency of the acoustic signal. Total path loss $A(l, f)$ [29] due to spreading and absorption can be expressed in decibels by the following equation:

$$A(l,f) = 10.\log(l^k) + \alpha(f) \times l \ (dB), \quad 4.1$$

where l (km) is the transmission range of the acoustic signal, k is the spreading factor, $\alpha(f)$ (in dB/km) is the absorption coefficient, and f (kHz) is the frequency. The value of k = 1 is for cylindrical spreading, and k = 2 for spherical spreading. A graph of absorption coefficient versus acoustic signal frequency is depicted in Figure 4-3 [92]. We can see that for frequencies up to 20 kHz, it is $\alpha\ (f) < 4$ dB/km (approximately).

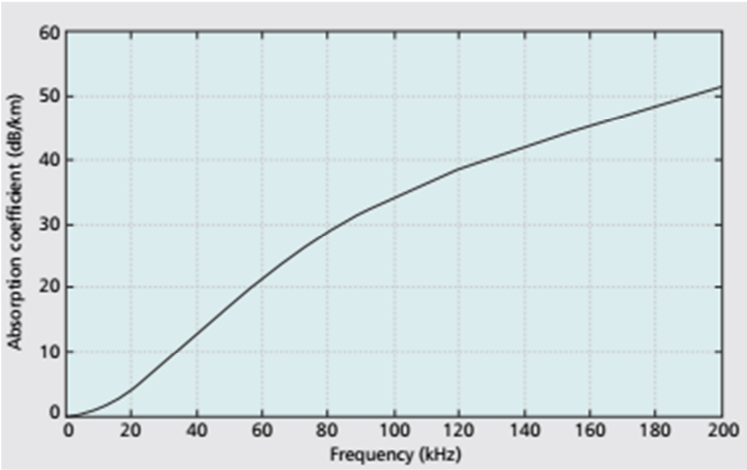


Figure 4-3. Absorption Coefficient vs Frequency.

Figure 4-4 [92] shows that Signal-to-noise-ratio (SNR) also depends on the frequency. This figure also shows that bandwidth of acoustic signal depends on the distance. For 100 km, the bandwidth is about 1 kHz, whereas for 5 km it is about 10 kHz.

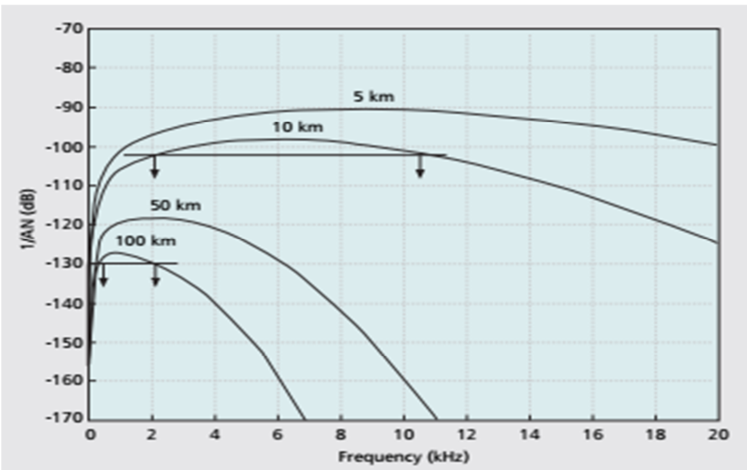


Figure 4-4. SNR vs Frequency and Distance.

Fading is more severe in shallow water compared to deep water due to multipath. There are no standard fading models for acoustic communication, and experimental measurements are used to predict the channel behavior [92]. For this protocol, the error in estimation of RSS due to losses, fading, and limited mobility of the nodes is not significant as long as a node successfully forms a path with one of its neighbors that is closer to the gateway. The error in node selection may be corrected later by measuring the RSS value at regular intervals using the data packets.

Our proposed protocol assumes that the nodes are deployed randomly in a 3D network. The minimum distance between nodes is 300 m and maximum is 1000 m. The transmission power is adaptable and can be adjusted according to the distance between the transmitter and sender. The nodes are not stationary and keep moving in all directions, although their movement is limited by the length of the binding lines to the surface buoy or the anchor at the sea bottom (see Figure 4-1). Lists of the acronyms and symbols used in this paper are shown in Table 4.1 and

Table 4.2, respectively.

Table 4.1 Packet Descriptions.

Packet Name	Description	Code
RR	Route Request	0000111
RR_ACK	Route Request ACKnowledgement	0100101
RR_RSP	RR ReSPonse	0101101

Table 4.2. List of the symbols used in the equations.

Symbol	Description
A	Absorption coefficient
F	Frequency in kHz
Ghwt	Number of neighbors weight
HP	Normalized number of hops multiplied by the its weight
Hpwt	Number of hops weight
NB	Normalized number of neighbors
NG	Normalized number of neighbors multiplied by the its weight
NH	Normalized number of hops
NS	Normalized Number of RSS
R	Distance between the nodes in meters
RS	Normalized RSS multiplied by the its weight
selnode	Selected node
Ss	RSS in dB
Sswt	RSS weight
TL	Transmission loss in dB
Y	Node selection equation

The process of route formation is initiated by the gateway node. The gateway broadcasts a route request (RR) packet, which will traverse throughout the network. When a node receives the RR packet, it measures the strength of the received signal. Initially, all the nodes will transmit at the same power, which will be enough to cover the maximum distance between the nodes. With the help of RSS, the distance between the RR sending node and receiving node is estimated. The RSS indicates the relative proximity between the sending node the receiving node. This is explained in the following example. Assume that three nodes, B, C, and H, are placed as illustrated in Figure 4-5.

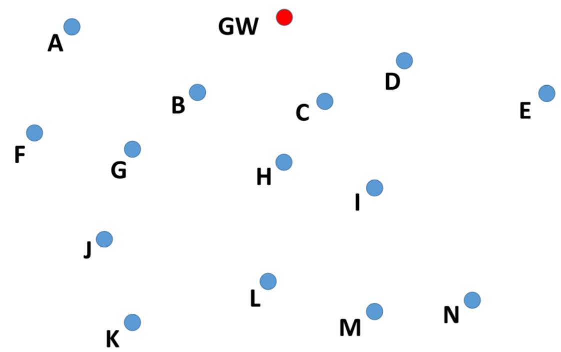


Figure 4-5. Non-uniformly deployed nodes in 2D network.

Suppose that nodes B and C have received the RR packet from the gateway. Nodes B and C will broadcast the RR packet at randomly selected time slots. Since H is the neighbor of both nodes, it will receive the RR packets from nodes B and C. When H receives the RR packet from B, it measures the RSS value and records it in a table. Similarly, H records the RSS value when it receives the RR packet from C. After that, it will compare the RSS value of both B and C, and will choose the node which has the stronger RSS. In this case, H will choose C as the next hop to forward the packet towards the gateway, because it has stronger RSS compare to B. The H node chooses the node having stronger RSS because it assumes that strong RSS is due to the shorter distance. However, before making the final decision, H will consider both RSS values, the number of hops between the candidate nodes and the gateway, and the number of neighbors of the candidate nodes.

In the case of a short distance between the sender and the receiver, less transmission energy is required by the sender to transmit the packet. Therefore, when the weight of the RSS is set to maximum (i.e., 1), the closer candidate node is selected. This saves energy at the sender node. However, this approach may decrease the throughput if too many hops are added in the path between the original packet sender and the gateway. Therefore, two more metrics are considered at the time of selection of the forwarding node: number of hops and number of neighbors. The first one is because the number of hops affects the energy consumption and throughput. The second one is because a high number of neighbors means a better chance to be selected as the forward node, so it is a good metric to be considered. To better understand the idea, a scenario is considered. Suppose that a node, V, has two candidate nodes, Q and M, as shown in Figure 4-6. Node Q is 300 m away from node V, has five neighbors (U, T, P, L, M) and can reach the gateway in four hops (L–H–B–GW), whereas node M is 600 m away from node V, has four neighbors (Q, L, H, N) and can reach the gateway in three hops (H–B–GW). At the time of forwarding node selection, node V will select the node based on all three metrics. The weights given to each metric will influence the selection accordingly. The selection is based on these three metrics to achieve a good balance between the throughput and the energy consumption. Suppose the shortest distance is given the highest weight, then node Q will be selected. If number of neighbors is given the highest weight, then node M will be selected, and if number of hops is given the highest weight then again M will be preferred over Q.

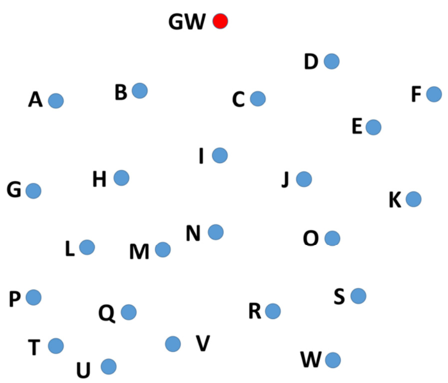


Figure 4-6. **Example of three metrics-distance, neighbors, and hops.**

The values of the metrics will be normalized and multiplied by their weights (from 0 to 1). Prior to the deployment of the nodes, the weights will be assigned for each metric of selection.

Packet collision at the receiver causes packet loss. The collision occurs when two or more nodes send packets to a node such that the packets arrive at the receiving node overlapped in time. All the nodes which can cause a collision in their transmission range form a collision domain, as shown in Figure 4-7. A network may have multiple collision domains. The number of nodes in a collision domain depends on the node's density and their transmission range. In order to avoid a collision, the nodes will send the RR packet at a randomly selected time slot from a set of possible time slots. Twenty time slots in the set are assumed to be sufficient for a moderately dense network. However, the number of slots may be increased in a densely populated network. The interval between two consecutive time slots will comprise packet delay, propagation delay, and guard time. For the latter, we assume that 5% of the total of packet delay and propagation delay will be enough to adjust the variance of the end to end packet transmission delay due to variations in the acoustic propagation velocity. When a node sends the RR packet, it will wait for the Route Request Acknowledgement (RR_ACK) packet from at least one node.

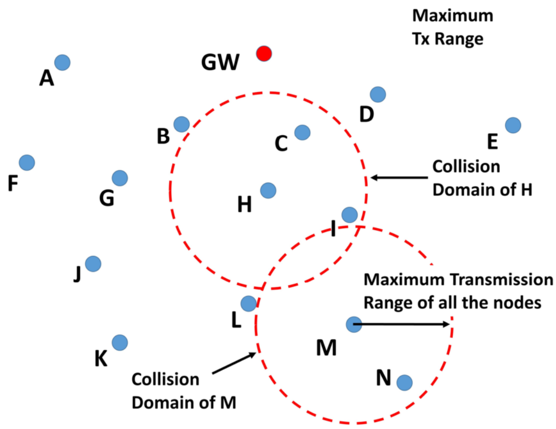


Figure 4-7. A Typical Collision Domain.

Once the RR packet has gone through all the nodes in the network, the nodes at end of the network send a Route Request Response RR_RSP packet back to the gateway. Nodes assume they are end nodes when they do not receive any response packet after sending the RR packet. When the gateway receives the RR_RSP packet from all the first hop nodes it assumes that all the nodes have successfully formed the path to the gateway. The nodes keep measuring the RSS of the received packets during the data packet transmission to optimize the path. If a node finds a closer neighbor compared to the existing forwarding node, then the former selects the latter as the next hop to forward the packets.

4.3 Route Formation Process

The route formation process starts with broadcasting the RR packet from the gateway, which is forwarded through all the network nodes. The steps for broadcasting the RR packet are described below.

4.3.1 Forwarding the RR Packet

1. GW broadcasts an RR packet three times with its node ID to ensure that all the neighbor nodes have received it;
2. The nodes that receive the RR packet form the path with the GW without considering RSS, because they are at one hop distance from the GW;

3. The nodes that receive the RR packet broadcast an RR packet at randomly selected time slot to avoid collision;

4. A node might receive the RR packet from only one node or multiple nodes:

   4.1. If it receives the RR packet from just one node, then it forms its path to the GW through that node;

   4.2. If it receives the RR from multiple nodes, then it compares the RSS values, hop counts to the GW, and number of neighbors of the candidate nodes. It will select the forwarding node according to the weights given to RSS, the least hop count, and least number of neighbors to form the path;

5. Once the forwarding node is selected, the acknowledgement of the RR packet (RR_ACK) is sent. An RR sending node receives acknowledgment from all nodes which are within its transmission range;

6. Steps 1–5 (broadcast of RR packet) are repeated by all the nodes that received the RR packet earlier, until all the nodes in the network have received the RR packet.

The RR packet is transmitted, at the most, three times, if acknowledgement is not received. If that occurs, then the node concludes that it is an end node and it will start sending a response (RR_RSP) packet that contains its own identification number and the identification numbers of the forwarding nodes. The RR_RSP packet is not broadcast, but sent to the node which was earlier selected by each node as a packet forwarding node.

4.4 Selection Criteria

The forwarding node can be selected based on RSS, number of hops, and the number of neighbors of the forwarding node. If the protocol is used for delay sensitive data, then the smaller number of hops will be given more weight. If energy conservation is the prime goal, then RSS and least number of neighbors will be given preference over the number of hops. Below, a scenario is considered to understand the process of selection in more detail.

1. Suppose the node A (Figure 4-8) broadcasts the RR packets which is received by the nodes B and C because they are in the transmission range of node A;

2. The information contained in the RR packet is shown in Figure 4-9. The RSS between node A and the gateway (GW) is 2 (assumed). Node A is one hop away from GW and A has no knowledge about its neighbors so far. Node A learns about its neighbors when it broadcasts the RR packet and receives the acknowledgement(s);

3. When nodes B and C send the acknowledgement (RR_ACK), node A learns that it has become part of the routing path of nodes B and C;

4. Now either B or C will broadcast the RR packet first (depends on the selected time slot);

5. Let us assume that node B will send the RR packet first. When B broadcasts RR packet it contains the information of signal strength between A and itself, number of hops it is away from the gateway through node A, and the number of neighbors (see Figure 4-10);

6. At this point C will compare the RSS, the least number of hops and the least number of neighbors to form the routing path with A or B.

7. Node C can choose its path to the gateway by two routes: C → A or C → B → A:

   7.1. If the weight coefficient defined for the least number of hops has a higher value than the other two weight coefficients (for RSS and number of hops), then node C will select node A as the forwarding node for minimum delay in the packet delivery;

   7.2. If the weight coefficient for RSS has a higher value, then C will select B as the forwarding node to save energy.

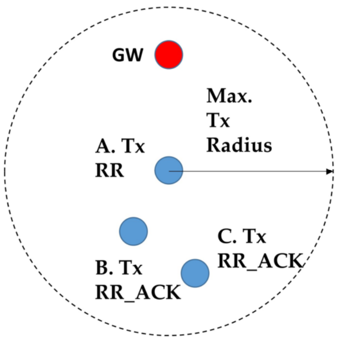


Figure 4-8. RR Transmission Range.

RR Broadcast From A				
From	To	RSS	Hops	Neighbors
A	GW	2	1	0
RSS value is assumed				

Figure 4-9. RR Packet Sent by Node A.

RR Broadcast From A				
From	To	RSS	Hops	Neighbors
A	GW	2	1	0
B	A	3	2	1
RSS values are assumed				

Figure 4-10. RR Packet Sent by Node B.

4.5 Mathematical Analysis

The criteria for path selection are quite straightforward. A node will prefer to form the routing path with the node that has the minimum distance, least number of hops, and least number of neighbors. The combined values of these parameters are calculated based on the weight given to each parameter. If transmitting power is to be conserved, then signal strength will be given the maximum weight, whereas if the higher throughput is the objective then the least number of hops will be given the maximum weight. Hence, when a node receives the RR packet from multiple nodes n it stores the signal strength of the received signal (ss, the number of hops between the gateway and the node which sent the RR packet (nh), and the number of neighbors (gh). We propose to use a new score function (n) evaluated for every node n to make the forwarding node selection. The expression of (n) is given as follows:

$$F(n) = (1 + RS(n)) * \left(1 - HP(n)\right) * (1 - NG(n)), \quad 4.2$$

where

$$RS(n\ ) = NS(n) * sswt,\ \{sswt \mid sswt \in \mathbb{R},\ 0 \leq sswt \leq 1\}, \quad 4.3$$

$$HP(n) = NH(n) * hpwt,\ \{hpwt \mid hpwt \in \mathbb{R},\ 0 \leq hpwt \leq 1\}, \quad 4.4$$

$$NG(n) = NB(n) * ghwt,\ \{ghwt \mid ghwt \in \mathbb{R},\ 0 \leq ghwt \leq 1\}, \quad 4.5$$

sswt, hpwt, and *ghwt* are the weights for RSS, hops, and neighbors, respectively. The sets of normalized values of RSS, number of hops, and number of neighbors are given by the following expressions, respectively:

$$NS = \{nss_1,\ nss_2,\ nss_3, \ldots,\ nss_n\}, \quad 4.6$$

$$NH = \{nhp_1,\ nhp_s,\ nhp_3, \ldots,\ nhp_n\}, \quad 4.7$$

$$NB = \{ngh_1,\ ngh_2,\ ngh_3, \ldots,\ ngh_n\}, \quad 4.8$$

A node i with a score (i) will be selected as the forwarding node when it fulfills the next rule:

$$F(i) = max\ \{F_1,\ F_2, F_3, \ldots, F_n\}, \quad 4.9$$

Received signal strength (ss) is calculated by:

$$ss = Txpwr - TL, \quad 4.10$$

where *Txpwr* is the transmitted signal power in logarithmic units, and *TL* is the transmission loss, also in dB, which is calculated by:

$$TL = \alpha R + 20\ log_{10}(R), \quad 4.11$$

where R is the distance between the nodes. The absorption coefficient α (dB/km) is calculated by [93]:

$$\alpha=\left(\frac{0.11f^{2}}{1+f^{2}}+\frac{44f^{2}}{4100f^{2}}\right)+2.75\times10^{-4}f^{2}+0.003, \qquad 4.12$$

where f is the frequency (kHz).

4.6 Packet Header Format

The header format of the RR, RR_ACK, and RR_RSP packets is shown in Figure 11. There are five header fields, S_ID, D_ID, Tx_Pwr, Seq._#, and Packet Type. The S_ID (source node identification number) and D_ID (destination node identification number) fields have 8 bits length. They are used by the nodes to identify each other. The Tx_Pwr (transmission power) field is used by the receiver node to estimate the path loss. The Seq._# field is used when a packet is transmitted multiple times, e.g., the RR packet is transmitted three times by the gateway. Finally, the receiver node identifies the type of packet received by the packet type field.

Bits	0	8	16	24	27	30
	S-ID (8)	D-ID (8)	Tx Pwr (8)	Seq. # (3)	Packet Type (3)	Data

Figure 4-11. Packet Header Format.

4.7 Computer Simulation Results

Simulations were carried out using MATLAB® to estimate two efficiency parameters: the average delay from sending nodes to the gateway, and the average number of hops that each node employed to reach the gateway. The simulation parameters are given in Table 4.3.

Table 4.3. Simulation Parameters.

Parameter	Value	Unit
Sound speed	1500	m/s
Data rate	5000	bits/s
Frequency	48	kHz
Header size	30	Bits
Transmission power	18	Watts
Nodes	100	nodes
Area	4x4	m^2
Depth	4	m
Nodes density	1.56	nodes/m^3
Number of runs	100	each

The average number of hops and delays were simulated for different weight combinations of RSS, number of hops, and number of neighbors. The starting weight was 0.2 and increased to 1 with steps of 0.2. The weight combinations used in the simulation are given in Table 4.4.

Table 4.4. Weights Combinations used in the Simulation.

	Weights														
RSS	0.2	0.4	0.6	0.8	1	0.4	0.3	0.2	0.1	0	0.4	0.3	0.2	0.1	0
Hops	0.4	0.3	0.2	0.1	0	0.2	0.4	0.6	0.8	1	0.4	0.3	0.2	0.1	0
Neighbors	0.4	0.3	0.2	0.1	0	0.4	0.3	0.2	0.1	0	0.2	0.4	0.6	0.8	1

The average number of hops for RSS weights 0.2 to 1 is shown in Figure 4-12. The graph shows that the average number of hops increased as RSS weight increased. This means that a node selects the forwarding node which closer to it, although it may have more nodes in the routing path to the gateway. The increase in the number of hops was approximately 34.83% as the RSS weight increased from 0.2 to 1 (6 versus 4.45 average hops).

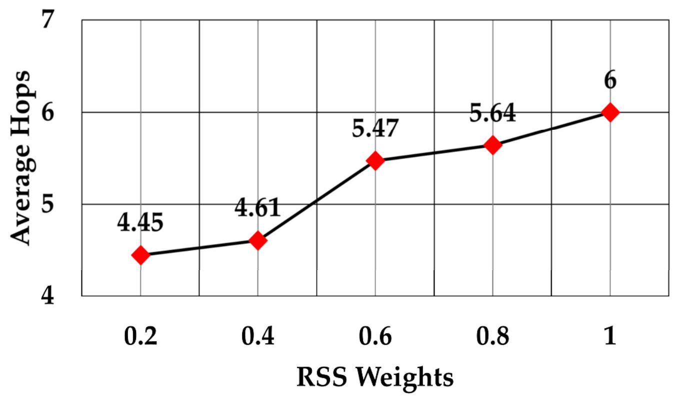


Figure 4-12. Number of Hops against RSS Weights.

The average number of hops using weights between 0.2 and 1 is shown in Figure 4-13. The graph shows how that figure decreased as the weight increased. This means that a node prefers to select the forwarding node which needs a smaller number of hops to reach the gateway. In this case, the throughput increased due to the smaller number of hops to reach the gateway, although this may increase the average energy consumption per node. Also, the average number of hops became almost constant after 60% weight of hops. Comparison of the average number of hops to reach the gateway when the RSS weight was the maximum (the other two weights were zero) and when the number of hops weight was the maximum (the other two weights were zero) shows that the average number of hops decreased by more than 30% (6 versus 4.16 average hops).

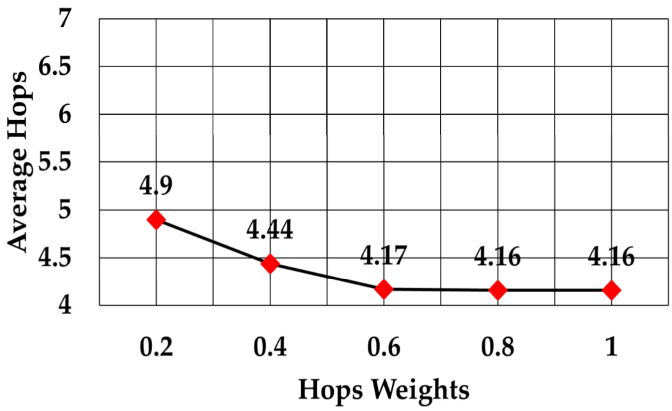


Figure 4-13. Number of Hops against Hops Weights.

When the weight for least number of neighbors went from 0.2 to 1, the graph shown in Figure 4-14 was obtained. Like in case of RSS, the average number of hops increased as the weight for least number of neighbors increased.

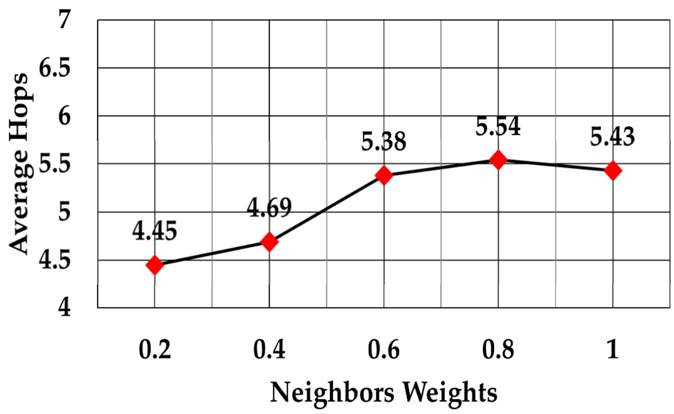


Figure 4-14. Number of Hops against Number of Neighbors Weights.

The analysis of packet delay from the node sending data to the gateway showed that it followed the same trend as the average number of hops, as can be observed in Figures Figure 4-15 to Figure 4-17. In the case of the least number of hops, the maximum weight decreased the delay by 12.2% compared to delay at weight = 0.2 (2.29 versus 2.61 min), 19.6% compared to delay at the maximum weight of RSS (2.29 versus 2.85 min), and 25.4% compared to delay at the maximum weight of least number of neighbors (2.29 versus 3.07 min).

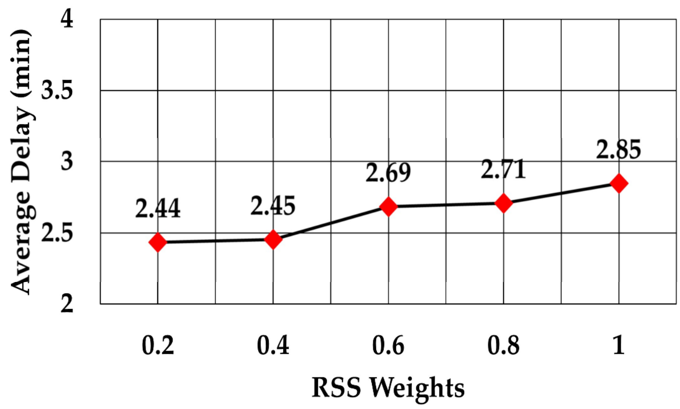


Figure 4-15. Delay against RSS Weights.

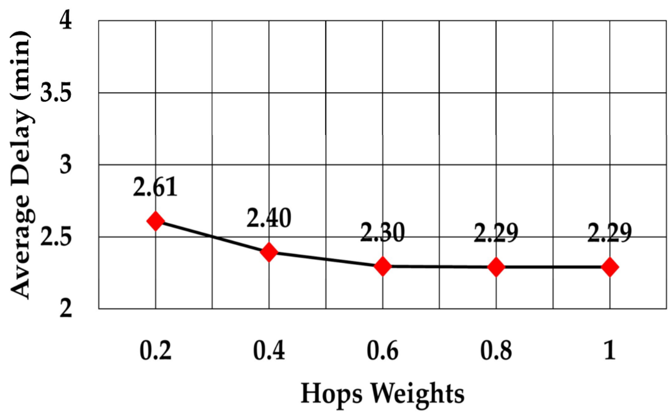


Figure 4-16. Delay against Hops Weights.

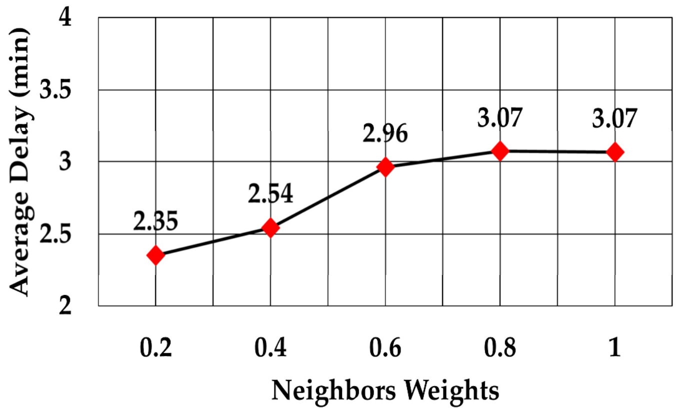


Figure 4-17. Delay against Number of Neighbor Nodes

The average energy consumption of a node to send a data packet was analyzed for multiple densities of the nodes. To compare with values given in RECRP [94] we assumed the same simulation parameters. The network area was 10 km × 10 km × 10 km and the number of nodes was 100, 200, 300, 400, 500, and 600. The data packet size was 256 bits and the transmission range was adjusted according to the node's density. The average energy consumption for multiple transmission range for each number of nodes was simulated as shown in Table 4.5. The simulation was run 10 times for each transmission range.

Table 4.5. Transmission Ranges for Different Number of Nodes.

S. No.	Nodes	Nodes Density	Tx. Ranges (km)
1	100	10	5, 6, 7, 8
2	200	5	4, 5, 6
3	300	3.33	3, 4, 5
4	400	2.5	2, 3, 4
5	500	2	2, 3
6	600	1.66	2, 3

The average energy consumption per node per packet for 100, 200, 300, 400, 500, and 600 nodes is shown in Figure 4-18, Figure 4-19, Figure 4-20, Figure 4-21, Figure 4-22 and Figure 4-23, respectively. As we expected, the energy consumption increased as the transmission range increased.

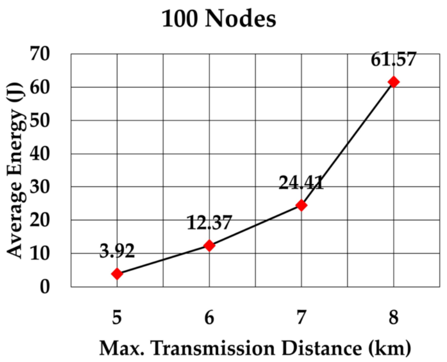


Figure 4-18. Energy Consumption per Node per Packet for 100 Nodes.

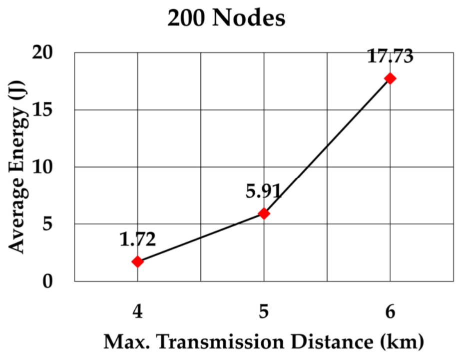


Figure 4-19. Energy Consumption per Node per Packet for 200 Nodes.

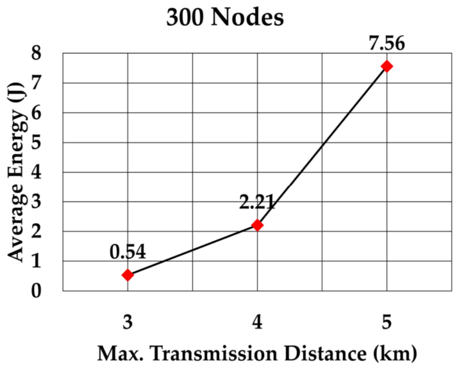


Figure 4-20. **Energy Consumption per Node per Packet for 300 Nodes.**

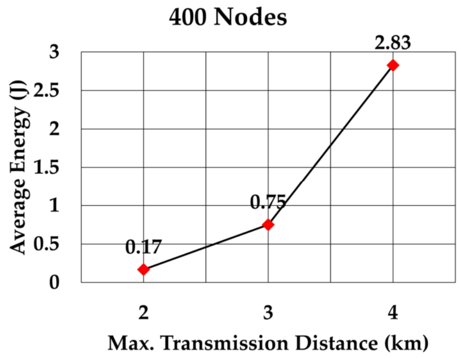


Figure 4-21. **Energy Consumption per Node per Packet for 400 Nodes.**

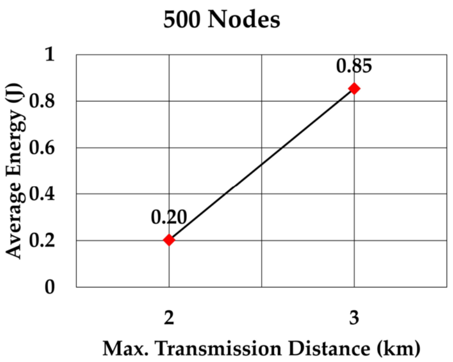


Figure 4-22. **Energy Consumption per Node per Packet for 500 Nodes.**

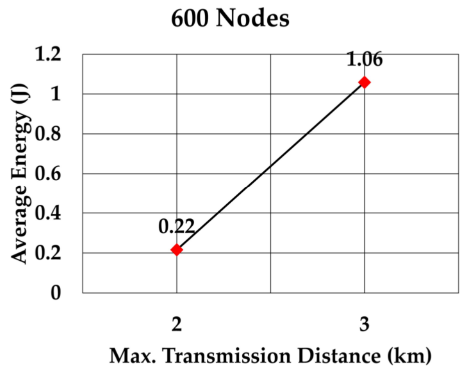


Figure 4-23. **Energy Consumption per Node per Packet for 600 Nodes.**

Our proposed protocol is very close to RECRP but there are subtle differences. RECRP is a reactive protocol, whereas our proposed protocol is based on proactive approach. One of the RECRP forwarding nodes selection metrics is the residual energy of the candidate nodes. Instead of residual energy, we considered the number of data packets which a candidate node will have to forward. If a candidate node is already selected as a forwarding node by many neighbor nodes, then it will increase the energy consumption of the forwarding node and decrease the throughput. In order to keep the data traffic balanced among the nodes, the number of neighbors for each node is also a metric for node selection. Our proposed protocol can be maximized in terms of transmission energy by selecting the nodes with the shortest distance. The number of relay nodes between the data sending node and the sink node increases the delay and decreases the throughput. Therefore, our proposed protocol can be maximized to increase the throughput by minimizing the relay nodes. We can also optimize the protocol for energy efficiency by selecting the appropriate weights for distance and number of neighbors, and in the case of throughput, by selecting the appropriate weights for number of hops.

The energy consumption of SPRINT is shown in Figure 4-24 and energy consumption by RECRP is shown in Figure 4-25 (taken from [94]). From the comparison of the two graphs, it is obvious that the energy consumption of SPRINT is much lower than RECRP under the same simulation conditions.

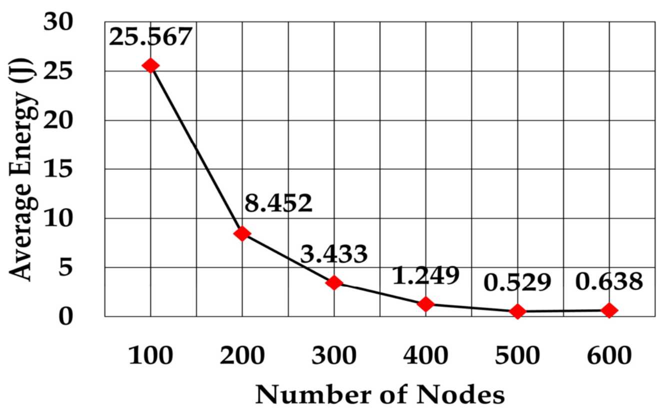


Figure 4-24. Average Energy Consumption.

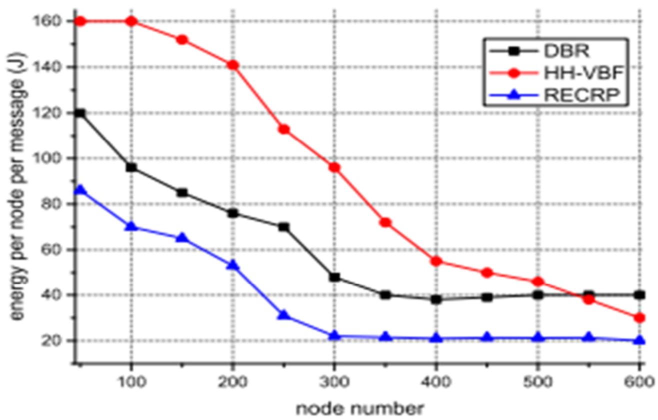


Figure 4-25. RECRP Energy Consumption.

The values for RECRP have been tabulated from Figure 4-25 (approximately) and compared to SPRINT simulations in Table 4.6.

Table 4.6. SPRINT and RECRP Energy Consumption Comparison.

	Energy Consumptions (J)		
Nodes	SPRINT	RECRP (approx.)	SPRINT Reduction (%)
100	25.567	70	63%
200	8.451	53	84%
300	3.432	22	84%
400	1.249	21.5	94%
500	0.528	21	97%
600	0.637	20	96%

4.8 Conclusions

In this paper, we presented a routing protocol for randomly deployed underwater network. The sensor nodes are deployed randomly to monitor the environment or to warn of natural disasters, like tsunamis. The protocol is designed to optimize the data throughput and energy consumption of the sensor nodes. This protocol does not require additional sensing devices to ascertain the location. Hence, the proposed routing protocol is based neither on the location of the nodes nor on the topology of the network.

SPRINT is a proactive protocol to minimize the routing delay. Each node that wants to send a packet knows the forwarding node in advance. At regular intervals, when a node receives the data packet, it computes RSS, hops, and neighbors to optimize the routing path and overcome the issue of a dead node. This regular update of the routing path makes the protocol resilient and more efficient.

However, computing routing path parameters upon receiving each data packet will increase the energy consumption significantly; therefore, the routing path update will be carried out after certain number of data packets, depending on the arrival rate.

Energy consumption is low due to the adaptive transmission power, which is adjusted with the help of RSS estimation. Throughput can also be increased by reducing the relay nodes between the data source node and the gateway. Data traffic among the relay nodes is distributed evenly by considering the number of neighbors at the time of forwarding node selection.

Chapter 5

Conclusions and Future Work

In this chapter we present the summary of our research work. First we explained the applications, network architectures and issues pertinent to UWSN in order to understand the reason behind this research work. As mentioned in the introduction chapter, the very first issue with underwater wireless communication is that electromagnetic waves cannot be used due to high absorption rate. The previous research and experience have shown that the acoustic wave is the most suitable alternative of EM wave. However, acoustic wave has an inherent characteristic of very slow propagation speed compared to EM wave. Typical speed of acoustic waves in the water is 1500 m/s. This slow propagation speed incurs substantial delay in packet transmission from one node to the other.

We know that UWSN is an extension of WSN, but we cannot use the WSN protocols readily for UWSN due to large propagation delay of acoustic waves. In addition to that, for time-critical applications, end-to-end packet delivery delay becomes an important factor. The propagation delay cannot be altered but the other delays can be minimized for time-critical applications. One such delay is the routing delay. There are two kinds of routing protocols namely reactive and proactive. We prefer reactive routing protocols for terrestrial WSN networks, like AODV [95] , because of better scalability and less routing overhead. But for time-critical applications in UWSN we prefer proactive protocols to minimize the route-finding delay.

When designing a routing protocol for UWSN we may have to take the network architecture into consideration as well. There are two UWSN architectures called 2D architecture and 3D architecture. In 2D architecture the node are fixed at the bottom of the sea whereas in 3D architecture the nodes are anchored to the bottom or attached to a surface buoy and keep float in the water. The nodes can also be deployed in a well-defined topology or non-uniformly.

Many routing protocols use time synchronization and location information to form the routing path. However, achieving the accurate time synchronization and location of the nodes in UWSN is very challenging. Therefore, routing protocols using timestamp or location of the node are prone to failure due to errors in time synchronization and localization.

In order to understand the strengths and weaknesses of the existing routing protocols, we classified them based on their working principle. We classified them and named them as location free, location based, opportunistic, cluster-based, energy-efficient, reliable data delivery and based on RSS. By in depth analysis of the existing routing protocols, we found some design issues in them. There are some routing protocols which are based on location information of the nodes and to get location information the use depth sensors. The use of any additional senor increases the energy consumption of the node which is an unwanted feature. There are routing protocols in which the routing paths are formed by the sink node after getting information from all the nodes. In this case the scalability and resilience are compromised because of centralized control. The summary of the issues is as given below.

1. Full duplex communication is difficult to implement in UWSN because of narrow acoustic bandwidth.
2. Determination of accurate location of the sensor nodes is very difficult.
3. Time synchronization is difficult to achieve in underwater networks because of variable-propagation delay and node mobility.
4. Use of additional devices like depth sensor and array of antennas (to measure the angle of arrival) increases the power consumption of the sensor node.
5. In centralized routing path formation, updating of the routing path in case of a node failure or addition of a node will not be easy. The updating process also decreases the data throughput and increases the energy consumption.
6. Reactive protocols are not suitable for time-critical applications because they add packet forwarding delay.
7. Cluster based protocols have relatively low throughput and the cluster heads become the bottle neck for the data traffic. Cluster based networks also create single point of failure.
8. Opportunistic routing protocols are good to increase the data delivery reliability, but they also increase the energy consumption of the nodes because multiple copies of the data packet are routed to the gateway.

As SOFRP is a self-organized protocol, we also needed a medium access method to transmit the RR and the other control packets. A mechanism of packet transmission at different times along with acknowledgement and retransmission is used to avoid the collision between the control packets. The nodes transmitted the control packets at a random time selected from a set of 40 time slots. However, the random selection couldn't have eliminated the chance of collision completely. Therefore, the mechanism of acknowledgment and retransmission was also added.

When the nodes are deployed, they have no information about the existence of the other nodes in the network and topology of the network. The gateway initiates the nodes and the routes discovery process by broadcasting the RR packet. The RR packet is propagated throughout the network to find the nodes in the network and form the routing path to the gateway. Each node forms routing path with the node above it (closer to the sea surface). Hence a string of nodes (virtually) is formed from the bottom of the sea up to the gateway.

The discovery process of the network nodes and the routes progress layer by layer through the network. The nodes find the neighbors at the same layer and form the routing path with the nodes above them. This process continues until the last node or nodes in the network are reached. In case of a node failure, the nodes form a new route with the adjacent string.

The performance of SOFRP is analyzed by simulating the protocol on MATLAB® for convergence delay, packets retransmission and collision rate. For convergence delay it is observed that, it remains the same when the number of strings increase from odd number to even number. However, when the number of nodes is increased in a string then the increase in convergence delay is linear. The percentage of retransmitted packets is 11-13% at 20% packet loss probability. The collision of the packets increases non-linearly as the number of strings increase. However, increase in the number of nodes in a string has no conclusive effect on the collision rate. The effect of number of time slots in a set is also analyzed in terms of collision rate and convergence delay. The purpose was to find the optimal number of time slots. The analysis shows that 30 (approx..) is the optimum value, however, we have chosen 40 time slots to reduce the number of collisions at the cost of little higher convergence delay. Comparison of SOFRP with NLPU shows that, 75% improvement in end-to-end packet delay was achieved by SOFRP.

SOFRP is designed for fixed radial topology networks which somehow limits it application where networks cannot be deployed in fixed radial topology. To overcome this limitation another protocol, called SPRINT, is designed for non-uniformly deployed networks with same objectives as SOFRP. SPRINT is also a proactive protocol which seeks the shortest path to decrease the routing delay. To achieve the shortest path a packet sending node selects the next nearest node. This requires a node to determine the distance between itself and all the single hop candidate nodes around it. The distance between the nodes is determined with the help of RSS. To achieve the shortest path the number of hops between the sender node and the gateway must also be the minimum. Therefore, the selection of the forwarding node also considers the number of hops required by the candidate nodes to reach the gateway. Another important factor in end-to-end latency is the traffic load on the packet forwarding nodes. Hence, to balance the traffic load and the energy consumption among the nodes, the forwarding node selection also considers the number of neighbors of each candidate node. So, the selection of forwarding node is based on three parameters namely RSS, number of hops and number of neighbors. Weights are assigned to the three parameters according to priority at the time of network deployment. If energy conservation is more important than maximum weight is given to RSS to select the nearest forwarding node. If minimum end-to-end delay is the priority, then the least number of hops is set to maximum.

Like SOFRP, the gateway initiates the discovery process of the network nodes and the routes. The RR packet is broadcast by the gateway which is received by all the neighbors within the transmission range of the gateway. At randomly selected time, all the nodes which received the RR packet from the gateway will broadcast the RR packet turn by turn. The nodes will keep propagating the RR packet until the farthest node away from the gateway reached. To optimize the routing paths and overcome the problem of node failures, the nodes will keep measuring the RSS using data packets. An appropriate interval for update is chosen based on data packets transmission rate. Adaptive transmission power for the data packets is also implemented to conserve the energy.

The simulation results show that the number of hops increase as the weight for RSS increases. It means that the end-to-end delay of packet delivery will also increase. Increasing the weight of least number of hops decreases the number of hops and increasing the least number of neighbors also increases the number of hops. Energy consumption decreases as the number of the nodes in a fixed network area increases. That is because of the distance between the nodes decreases as the number of nodes increases.

The weights for the three parameters in SPRINT are initially fixed at the time of deployment. The performance of the network in terms of throughput and energy conservation can be enhanced if the parameters can be adjusted according to the network and channel conditions. In future we can work on making the weights of the three parameters automatically adjustable according to the network conditions.

5.1 Objectives reached in this Thesis

Most of the existing protocols depend on time synchronization and location information which either makes them inefficient or difficult to implement. Opportunistic routing protocols have inherent issue of redundant packets which decreases the throughput and increases the energy consumption. Cluster-based protocols are not suitable for time critical application because of the data congestion at the cluster head. The reactive protocols which use handshaking mechanism are also not suitable for time critical applications. Therefore, we designed a protocol which is proactive, self-organized, cross-layered, resilient, non-cluster-based and does not require time synchronization, location information and, additional device like depth sensor which makes it more practical protocol for time critical applications.

5.2 Future work

In future we need to develop the networks in which the sensor nodes are not deployed in a fixed area. Such networks may be required by the military for mobile surveillance or by oceanographers for environmental monitoring. Therefore, we intend to design a routing protocol for underwater mobile adhoc networks (UWMAN). UWMAN will be comprised of freely moving sensor nodes like AUVs or ROVs. The continuously changing position of the sensor nodes makes the packet routing and communication among the nodes quite challenging. Our goal would be to design UWMAN having the characteristics similar to SOFRP and SPRINT.

References

[1] Chee-Yee Chong and S. P. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003, doi: 10.1109/JPROC.2003.814918.

[2] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next century challenges: mobile networking for 'Smart Dust,'" in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking - MobiCom '99*, Seattle, Washington, United States, 1999, pp. 271–278, doi: 10.1145/313451.313558.

[3] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 72–83, Jan. 2000, doi: 10.1109/48.820738.

[4] "Dynamic Behavior of a Semi-Submersible Platform Coupled With Drilling Riser During Re-Entry Operation in Ultra-Deep Water," *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/267604275_Dynamic_Behavior_of_a_Semi-Submersible_Platform_Coupled_With_Drilling_Riser_During_Re-Entry_Operation_in_Ultra-Deep_Water. [Accessed: 14-Dec-2019].

[5] P. An, N. Willis, and S. Hatton, "Standalone Subsea Data Monitoring System," in *6th Underwater Science Symposium*, Aberdeen University, Scotland, 2003, p. 23.

[6] N. Mohamed, I. Jawhar, J. Al-Jaroodi, and L. Zhang, "Sensor Network Architectures for Monitoring Underwater Pipelines," *Sensors*, vol. 11, no. 11, pp. 10738–10764, Nov. 2011, doi: 10.3390/s111110738.

[7] D. Whang, N. Xu, S. Rangwala, K. Chintalapudi, R. Govindan, and J. W. Wallace, "Development of an Embedded Networked Sensing System for Structural Health Monitoring," in *Center for Embedded Network Sensing*, 2004.

[8] A. Cerpa, J. Elson, D. Estrin, L. Girod, M. Hamilton, and J. Zhao, "Habitat monitoring: application driver for wireless communications technology," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 2 supplement, p. 20, Apr. 2001, doi: 10.1145/844193.844196.

[9] A. Quazi and W. Konrad, "Underwater acoustic communications," *IEEE Commun. Mag.*, vol. 20, no. 2, pp. 24–30, Mar. 1982, doi: 10.1109/MCOM.1982.1090990.

[10] J. Lloret, S. Sendra, M. Ardid, and J. J. P. C. Rodrigues, "Underwater Wireless Sensor Communications in the 2.4 GHz ISM Frequency Band," *Sensors*, vol. 12, no. 4, pp. 4237–4264, Mar. 2012, doi: 10.3390/s120404237.

[11] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.*, vol. 370, no. 1958, pp. 158–175, Jan. 2012, doi: 10.1098/rsta.2011.0214.

[12] K. Nakayachi and K. Nagaya, "The Effects of the Passage of Time from the 2011 Tohoku Earthquake on the Public's Anxiety about a Variety of Hazards," *Int. J. Environ. Res. Public. Health*, vol. 13, p. 866, Aug. 2016, doi: 10.3390/ijerph13090866.

[13] "Oil/Gas Pipeline Leak Inspection and Repair in Underwater Poor Visibility Conditions: Challenges and Perspectives." [Online]. Available: https://www.scirp.org/journal/PaperInformation.aspx?PaperID=19510. [Accessed: 15-Apr-2019].

[14] S. Kemna, M. J. Hamilton, D. T. Hughes, and K. D. LePage, "Adaptive autonomous underwater vehicles for littoral surveillance," *Intell. Serv. Robot.*, vol. 4, no. 4, p. 245, Jul. 2011, doi: 10.1007/s11370-011-0097-4.

[15] A. Caiti, V. Calabrò, A. Munafò, G. Dini, and A. L. Duca, "Mobile Underwater Sensor Networks for Protection and Security: Field Experience at the UAN11 Experiment," *J. Field Robot.*, vol. 30, no. 2, pp. 237–253, 2013, doi: 10.1002/rob.21447.

[16] "GITEWS: Status." [Online]. Available: https://www.gitews.org/en/status/. [Accessed: 16-Jan-2020].

[17] N. O. and A. A. US Department of Commerce, "National Data Buoy Center." [Online]. Available: https://www.ndbc.noaa.gov/. [Accessed: 16-Jan-2020].

[18] N. G. Society, "Save the Plankton, Breathe Freely," *National Geographic Society*, 28-Feb-2012. [Online]. Available: http://www.nationalgeographic.org/activity/save-the-plankton-breathe-freely/. [Accessed: 15-Apr-2019].

[19] "Survey of the State of the World's Lakes," *ILEC*. [Online]. Available: https://www.ilec.or.jp/en/. [Accessed: 15-Apr-2019].

[20] G. Acar and A. E. Adams, "ACMENet: an underwater acoustic sensor network protocol for real-time environmental monitoring in coastal areas," *Sonar Navig. IEE Proc. - Radar*, vol. 153, no. 4, pp. 365–380, Aug. 2006, doi: 10.1049/ip-rsn:20045060.

[21] C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A Robust, Adaptive, Solar-Powered WSN Framework for Aquatic Environmental Monitoring," *IEEE Sens. J.*, vol. 11, no. 1, pp. 45–55, Jan. 2011, doi: 10.1109/JSEN.2010.2051539.

[22] S. Zhang, J. Yu, A. Zhang, L. Yang, and Y. Shu, "Marine vehicle sensor network architecture and protocol designs for ocean observation," *Sensors*, vol. 12, no. 1, pp. 373–390, 2012, doi: 10.3390/s120100373.

[23] C. C. Leroy and F. Parthiot, "Depth-pressure relationships in the oceans and seas," *J. Acoust. Soc. Am.*, vol. 103, no. 3, pp. 1346–1352, Mar. 1998, doi: 10.1121/1.421275.

[24] H. Medwin, "Speed of sound in water: A simple equation for realistic parameters," *J. Acoust. Soc. Am.*, vol. 58, no. 6, pp. 1318–1319, Dec. 1975, doi: 10.1121/1.380790.

[25] C. Chen and F. J. Millero, "Speed of sound in seawater at high pressures," *J. Acoust. Soc. Am.*, vol. 62, no. 5, pp. 1129–1135, Nov. 1977, doi: 10.1121/1.381646.

[26] H. Sarker, Md. Oli-Uz-Zaman, I. H. Chowdhury, S. A. H. Chowdhury, and Md. R. Islam, "Node Estimation Approach of Underwater Communication Networks Using Cross-correlation for Direct and Multi-path Propagation," in *2019 International Conference on Robotics,Electrical and Signal Processing Techniques (ICREST)*, 2019, pp. 286–291, doi: 10.1109/ICREST.2019.8644336.

[27] M. A. Ainslie, "Effect of wind-generated bubbles on fixed range acoustic attenuation in shallow water at 1–4kHz," *J. Acoust. Soc. Am.*, vol. 118, no. 6, pp. 3513–3523, Dec. 2005, doi: 10.1121/1.2114527.

[28] M. Stojanovic, "On the Relationship Between Capacity and Distance in an Underwater Acoustic Communication Channel," p. 7.

[29] M. Stojanovic, "On the Relationship Between Capacity and Distance in an Underwater Acoustic Communication Channel," in *Proceedings of the 1st ACM International Workshop on Underwater Networks*, New York, NY, USA, 2006, pp. 41–47, doi: 10.1145/1161039.1161049.

[30] X. Lurton, *An Introduction to Underwater Acoustics: Principles and Applications*, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2010.

[31] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 257–279, May 2005, doi: 10.1016/j.adhoc.2005.01.004.

[32] W. Wang, H. Huang, F. He, F. Xiao, X. Jiang, and C. Sha, "An Enhanced Virtual Force Algorithm for Diverse k-Coverage Deployment of 3D Underwater Wireless Sensor Networks," *Sensors*, vol. 19, no. 16, p. 3496, Jan. 2019, doi: 10.3390/s19163496.

[33] J. U. Khan and H. Cho, "A data gathering protocol using AUV in underwater sensor networks," in *OCEANS 2014 - TAIPEI*, 2014, pp. 1–6, doi: 10.1109/OCEANS-TAIPEI.2014.6964549.

[34] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges," *Ad Hoc Netw.*, vol. 3, pp. 257–279, May 2005, doi: 10.1016/j.adhoc.2005.01.004.

[35] D. Pompili, T. Melodia, and I. F. Akyildiz, "Distributed Routing Algorithms for Underwater Acoustic Sensor Networks," *IEEE Trans. Wirel. Commun.*, vol. 9, no. 9, pp. 2934–2944, Sep. 2010, doi: 10.1109/TWC.2010.070910.100145.

[36] Daeyoup Hwang and Dongkyun Kim, "DFR: Directional flooding-based routing protocol for underwater sensor networks," in *OCEANS 2008*, Quebec City, QC, Canada, 2008, pp. 1–7, doi: 10.1109/OCEANS.2008.5151939.

[37] Y. Noh *et al.*, "HydroCast: Pressure Routing for Underwater Sensor Networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 333–347, Jan. 2016, doi: 10.1109/TVT.2015.2395434.

[38] M. C. Domingo and R. Prior, "A Distributed Clustering Scheme for Underwater Wireless Sensor Networks," in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2007, pp. 1–5, doi: 10.1109/PIMRC.2007.4394038.

[39] W. Liang, H. Yu, L. Liu, B. Li, and C. Che, "Information-Carrying Based Routing Protocol for Underwater Acoustic Sensor Network," in *2007 International Conference on Mechatronics and Automation*, 2007, pp. 729–734, doi: 10.1109/ICMA.2007.4303634.

[40] G. G. Xie and J. H. Gibson, "A network layer protocol for UANs to address propagation delay induced performance limitations," in *MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No.01CH37295)*, 2001, vol. 4, pp. 2087–2094 vol.4, doi: 10.1109/OCEANS.2001.968320.

[41] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in Protocol Research for Underwater Acoustic Sensor Networks," in *Proceedings of the 1st ACM International Workshop on Underwater Networks*, New York, NY, USA, 2006, pp. 7–16, doi: 10.1145/1161039.1161043.

[42] M. Barbeau, S. Blouin, G. Cervera, J. Garcia-Alfaro, and E. Kranakis, "Location-free link state routing for underwater acoustic sensor networks," in *2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*, Halifax, NS, Canada, 2015, pp. 1544–1549, doi: 10.1109/CCECE.2015.7129510.

[43] A. Khan *et al.*, "A Localization-Free Interference and Energy Holes Minimization Routing for Underwater Wireless Sensor Networks," *Sensors*, vol. 18, no. 2, p. 165, Jan. 2018, doi: 10.3390/s18010165.

[44] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks," in *NETWORKING 2006. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, vol. 3976, F. Boavida, T. Plagemann, B. Stiller, C. Westphal, and E. Monteiro, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 1216–1221.

[45] P. Xie, Z. Zhou, N. Nicolaou, A. See, J.-H. Cui, and Z. Shi, "Efficient Vector-Based Forwarding for Underwater Sensor Networks," *EURASIP J. Wirel. Commun. Netw.*, vol. 2010, no. 1, p. 195910, Dec. 2010, doi: 10.1155/2010/195910.

[46] H. Yu, N. Yao, and J. Liu, "An Adaptive Routing Protocol in Underwater Sparse Acoustic Sensor Networks," *Ad Hoc Netw*, vol. 34, no. C, pp. 121–143, Nov. 2015, doi: 10.1016/j.adhoc.2014.09.016.

[47] X. Du, K. Huang, S. Lan, Z. Feng, and F. Liu, "LB-AGR: level-based adaptive geo-routing for underwater sensor network," *J. China Univ. Posts Telecommun.*, vol. 21, no. 1, pp. 54–59, Feb. 2014, doi: 10.1016/S1005-8885(14)60268-5.

[48] J. M. Jornet, M. Stojanovic, and M. Zorzi, "Focused beam routing protocol for underwater acoustic networks," in *Proceedings of the third ACM international workshop on Wireless network testbeds, experimental evaluation and characterization - WuWNeT '08*, San Francisco, California, USA, 2008, p. 75, doi: 10.1145/1410107.1410121.

[49] N. Chirdchoo, W. Soh, and K. C. Chua, "Sector-Based Routing with Destination Location Prediction for Underwater Mobile Networks," in *2009 International Conference on Advanced Information Networking and Applications Workshops*, 2009, pp. 1148–1153, doi: 10.1109/WAINA.2009.105.

[50] J. Shen, J. Wang, J. Wang, J. Zhang, and S. Wang, "Location-Aware Routing Protocol for Underwater Sensor Networks," in *Advanced Technologies, Embedded and Multimedia for Human-centric Computing*, 2014, pp. 609–617.

[51] M. Ayaz and A. Abdullah, "Hop-by-Hop Dynamic Addressing Based (H2-DAB) Routing Protocol for Underwater Wireless Sensor Networks," in *2009 International Conference on Information and Multimedia Technology*, 2009, pp. 436–441, doi: 10.1109/ICIMT.2009.70.

[52] H. Kim and H.-S. Cho, "SOUNET: Self-Organized Underwater Wireless Sensor Network," *Sensors*, vol. 17, no. 2, p. 283, Feb. 2017, doi: 10.3390/s17020283.

[53] S. Gopi, G. Kannan, D. Chander, U. B. Desai, and S. N. Merchant, "PULRP: Path Unaware Layered Routing Protocol for Underwater Sensor Networks," in *2008 IEEE International Conference on Communications*, Beijing, China, 2008, pp. 3141–3145, doi: 10.1109/ICC.2008.591.

[54] V. Chandrasekhar and W. Seah, "An Area Localization Scheme for Underwater Sensor Networks," in *OCEANS 2006 - Asia Pacific*, 2006, pp. 1–8, doi: 10.1109/OCEANSAP.2006.4393969.

[55] H. Yan, Z. J. Shi, and J.-H. Cui, “DBR: Depth-Based Routing for Underwater Sensor Networks,” in *NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, 2008, pp. 72–86.

[56] H. Yu, N. Yao, T. Wang, G. Li, Z. Gao, and G. Tan, “WDFAD-DBR: Weighting depth and forwarding area division DBR routing protocol for UASNs,” *Ad Hoc Netw.*, vol. 37, pp. 256–282, Feb. 2016, doi: 10.1016/j.adhoc.2015.08.023.

[57] “Multi-sink opportunistic routing protocol for underwater mesh network,” in *2008 International Conference on Communications, Circuits and Systems*, 2008, pp. 405–409, doi: 10.1109/ICCCAS.2008.4657802.

[58] Y. Noh, U. Lee, P. Wang, B. S. C. Choi, and M. Gerla, “VAPR: Void-Aware Pressure Routing for Underwater Sensor Networks,” *IEEE Trans. Mob. Comput.*, vol. 12, no. 5, pp. 895–908, May 2013, doi: 10.1109/TMC.2012.53.

[59] M. C.-C. Hung, K. C.-J. Lin, C.-F. Chou, and C.-C. Hsu, “EFFORT: energy-efficient opportunistic routing technology in wireless sensor networks,” *Wirel. Commun. Mob. Comput.*, vol. 13, no. 8, pp. 760–773, 2013, doi: 10.1002/wcm.1140.

[60] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, “An Opportunistic Void Avoidance Routing Protocol for Underwater Sensor Networks,” in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, 2016, pp. 316–323, doi: 10.1109/AINA.2016.96.

[61] Z. Jin, M. Ding, and S. Li, “An Energy-Efficient and Obstacle-Avoiding Routing Protocol for Underwater Acoustic Sensor Networks,” *Sensors*, vol. 18, no. 12, p. 4168, Nov. 2018, doi: 10.3390/s18124168.

[62] F. Ahmed, Z. Wadud, N. Javaid, N. Alrajeh, M. Alabed, and U. Qasim, “Mobile Sinks Assisted Geographic and Opportunistic Routing Based Interference Avoidance for Underwater Wireless Sensor Network,” *Sensors*, vol. 18, no. 4, p. 1062, Apr. 2018, doi: 10.3390/s18041062.

[63] W. Bai, H. Wang, K. He, and R. Zhao, “Path Diversity Improved Opportunistic Routing for Underwater Sensor Networks,” *Sensors*, vol. 18, no. 4, p. 1293, Apr. 2018, doi: 10.3390/s18041293.

[64] P. Wang, C. Li, and J. Zheng, “Distributed Minimum-Cost Clustering Protocol for UnderWater Sensor Networks (UWSNs),” in *2007 IEEE International Conference on Communications*, 2007, pp. 3510–3515, doi: 10.1109/ICC.2007.580.

[65] M. Ayaz, A. Abdullah, and L. T. Jung, “Temporary cluster based routing for Underwater Wireless Sensor Networks,” in *2010 International Symposium on Information Technology*, 2010, vol. 2, pp. 1009–1014, doi: 10.1109/ITSIM.2010.5561598.

[66] J. Jiang, G. Han, H. Guo, L. Shu, and J. J. P. C. Rodrigues, “Geographic multipath routing based on geospatial division in duty-cycled underwater wireless sensor networks,” *J. Netw. Comput. Appl.*, vol. 59, pp. 4–13, Jan. 2016, doi: 10.1016/j.jnca.2015.01.005.

[67] J. Chen, X. Wu, and G. Chen, “REBAR: A Reliable and Energy Balanced Routing Algorithm for UWSNs,” in *2008 Seventh International Conference on Grid and Cooperative Computing*, Shenzhen, Guangdong, China, 2008, pp. 349–355, doi: 10.1109/GCC.2008.12.

[68] Chun-Hao Yang and Kuo-Feng Ssu, “An energy-efficient routing protocol in underwater sensor networks,” in *2008 3rd International Conference on Sensing Technology*, Taipei, Taiwan, 2008, pp. 114–118, doi: 10.1109/ICSENST.2008.4757084.

[69] C.-J. Huang, Y.-W. Wang, H.-H. Liao, C.-F. Lin, K.-W. Hu, and T.-Y. Chang, "A power-efficient routing protocol for underwater wireless sensor networks," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2348–2355, Mar. 2011, doi: 10.1016/j.asoc.2010.08.014.

[70] A. Wahid, S. Lee, H.-J. Jeong, and D. Kim, "EEDBR: Energy-Efficient Depth-Based Routing Protocol for Underwater Wireless Sensor Networks," in *Advanced Computer Science and Information Technology*, vol. 195, T. Kim, H. Adeli, R. J. Robles, and M. Balitanas, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 223–234.

[71] A. Wahid, S. Lee, and D. Kim, "A reliable and energy-efficient routing protocol for underwater wireless sensor networks," *Int. J. Commun. Syst.*, vol. 27, no. 10, pp. 2048–2062, 2014, doi: 10.1002/dac.2455.

[72] C.-H. Yang, K.-F. Ssu, and C.-L. Yang, "A collision-analysis-based energy-efficient routing protocol in 3D underwater acoustic sensor networks," *Comput. Commun.*, vol. 66, pp. 25–35, Jul. 2015, doi: 10.1016/j.comcom.2015.04.004.

[73] S. Basagni, C. Petrioli, R. Petroccia, and D. Spaccini, "CARP: A Channel-aware routing protocol for underwater acoustic wireless networks," *Ad Hoc Netw.*, vol. 34, pp. 92–104, Nov. 2015, doi: 10.1016/j.adhoc.2014.07.014.

[74] M. M. V. Priya and A. Kumari, "Traffic aware multipath communication for time-critical applications in underwater acoustic sensor networks," *Int J Manag IT Eng*, vol. 2, pp. 66–73, Jan. 2012.

[75] E. Magistretti, J. Kong, U. Lee, M. Gerla, P. Bellavista, and A. Corradi, "A Mobile Delay-Tolerant Approach to Long-Term Energy-Efficient Underwater Sensor Networking," in *2007 IEEE Wireless Communications and Networking Conference*, 2007, pp. 2866–2871, doi: 10.1109/WCNC.2007.531.

[76] Y.-S. Chen and Y.-W. Lin, "Mobicast Routing Protocol for Underwater Sensor Networks," *IEEE Sens. J.*, vol. 13, no. 2, pp. 737–749, Feb. 2013, doi: 10.1109/JSEN.2012.2226877.

[77] P. Sun, W. K. G. Seah, and P. W. Q. Lee, "Efficient Data Delivery with Packet Cloning for Underwater Sensor Networks," in *2007 Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies*, 2007, pp. 34–41, doi: 10.1109/UT.2007.370944.

[78] D. Pompili, T. Melodia, and I. F. Akyildiz, "A Resilient Routing Algorithm for Long-term Applications in Underwater Sensor Networks," presented at the IFIP Med-Hoc-Net, 2006, p. 8.

[79] A. Wahid, S. Lee, and D. Kim, "A reliable and energy-efficient routing protocol for underwater wireless sensor networks," *Int J Commun. Syst.*, vol. 27, pp. 2048–2062, 2014, doi: 10.1002/dac.2455.

[80] G. Xie, "A Networking Protocol for Underwater Acoustic Networks," *Nav. Postgrad. Sch. CS Dept Monterey Calif. US*, 2000.

[81] K. Ali and H. Hassanein, "Underwater Wireless Hybrid Sensor Networks," in *2008 IEEE Symposium on Computers and Communications*, 2008, pp. 1166–1171, doi: 10.1109/ISCC.2008.4625635.

[82] X. Xerandy, T. Znati, and L. K. Comfort, "Cost-Effective, Cognitive Undersea Network for Timely and Reliable Near-Field Tsunami Warning," *Int. J. Adv. Comput. Sci. Appl. IJACSA*, vol. 6, no. 7, 2015, doi: 10.14569/IJACSA.2015.060730.

[83] K. Kawaguchi, Y. Kaneda, and E. Araki, "The DONET: A real-time seafloor research infrastructure for the precise earthquake and tsunami monitoring," in *OCEANS 2008 - MTS/IEEE Kobe Techno-Ocean*, 2008, pp. 1–4, doi: 10.1109/OCEANSKOBE.2008.4530918.

[84] Y. Zhang, "Wireless Sensor Network for Volcano Monitoring," *KTH K. Tek. Högsk. R. Inst. Technol. Stockh. Swed.*, 2005.

[85] W. A. Thelen, "Seismic instrumentation plan for the Hawaiian Volcano Observatory," U.S. Geological Survey, Reston, VA, USGS Numbered Series 2014–5179, 2014.

[86] "S2C R 48/78 | EvoLogics," *EvoLogics GmbH*. [Online]. Available: https://evologics.de/acoustic-modem/48-78/r-serie. [Accessed: 09-Apr-2019].

[87] M. Hosseini, H. Chizari, Chai Kok Soon, and R. Budiarto, "RSS-based distance measurement in Underwater Acoustic Sensor Networks: An application of the Lambert W function," in *2010 4th International Conference on Signal Processing and Communication Systems*, 2010, pp. 1–4, doi: 10.1109/ICSPCS.2010.5709656.

[88] D. R. K M, S.-H. Yum, E. Ko, S.-Y. Shin, J.-I. Namgung, and S.-H. Park, "Multi-Media and Multi-Band Based Adaptation Layer Techniques for Underwater Sensor Networks," *Appl. Sci.*, vol. 9, no. 15, p. 3187, Jan. 2019, doi: 10.3390/app9153187.

[89] H. Karl and A. Willig, "Localization and Positioning," in *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & Sons, Ltd, 2006, pp. 231–249.

[90] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks," in *NETWORKING 2006. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, 2006, pp. 1216–1221.

[91] S. Climent, A. Sanchez, J. V. Capella, N. Meratnia, and J. J. Serrano, "Underwater Acoustic Wireless Sensor Networks: Advances and Future Trends in Physical, MAC and Routing Layers," *Sensors*, vol. 14, no. 1, pp. 795–833, Jan. 2014, doi: 10.3390/s140100795.

[92] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 84–89, Jan. 2009, doi: 10.1109/MCOM.2009.4752682.

[93] L. M. Brekhovskikh and Y. P. Lysanov, *Fundamentals of Ocean Acoustics*, 3rd ed. New York: Springer-Verlag, 2003.

[94] J. Liu, M. Yu, X. Wang, Y. Liu, X. Wei, and J. Cui, "RECRP: An Underwater Reliable Energy-Efficient Cross-Layer Routing Protocol," *Sensors*, vol. 18, no. 12, Nov. 2018, doi: 10.3390/s18124148.

[95] I. D. Chakeres and E. M. Belding-Royer, "AODV routing protocol implementation design," in *24th International Conference on Distributed Computing Systems Workshops, 2004. Proceedings.*, 2004, pp. 698–703, doi: 10.1109/ICDCSW.2004.1284108.

Summary of Author's CV

Experience

- Assistant Professor: Computer Science Department at Ilma University – from 2019 -Current
- Assistant Professor: Electrical Engineering Department at Mohammad Ali Jinnah University (MAJU) – from 2013 to 2019.
- Cathodic Protection Engineer / CEO at Apex Engineering Services 2006 – 2013.
- Cathodic Protection Engineer in Al-Otaishan Company KSA
- Network administrator at Mega and In Company 2001 – 2002
- Network administrator at Karachi Institute of Information Technology (KIIT), 1998-1999

Education

- MS Telecommunication and Networking, Mohammad Ali Jinnah University, 2012
- BS Electrical and Electronics Engineering, Near East University, Cyprus, 1997

Journal Publications

- Hyder, Waheeduddin; Luque-Nieto, Miguel-Ángel; Poncela, Javier; Otero, Pablo. 2019. "Self-Organized Proactive Routing Protocol for Non-Uniformly Deployed Underwater Networks." Sensors (ISSN: 1424-8220), no. 24:5487, 2019. DOI: 10.3390/s19245487.
- Hindu, Sateesh K.; Hyder, Waheeduddin; Luque-Nieto, Miguel-Angel; Poncela, Javier; Otero, Pablo. 2019. "Self-Organizing and Scalable Routing Protocol (SOSRP) for Underwater Acoustic Sensor Networks." Sensors (ISSN: 1424-8220), no. 14: 3130, 2019. DOI: 10.3390/s19143130.
- Hyder, Waheeduddin; Poncela, Javier; Luque, Miguel-Angel; Otero, Pablo. 2018. "Self-organized Fast Routing Protocol for Radial Underwater Networks." Sensors (ISSN: 1424-8220), no. 12:4178, 2018. DOI: 10.3390/s18124178.
- Hyder, Waheeduddin; Nadeem, Adnan; Basit, Abdul; Rizwan, Kashif; Ahsan, Kamran; Mehmood, Nadeem. A Review and Classification of Energy Efficient MAC Protocols for Underwater Wireless Sensor Network. Journal of Basic & Applied Sciences, 2017, 13,63-70.

Conference Publications

- Waheeduddin, Hyder, Miguel-Angel, Luque-Nieto, Javier, Poncela, Pablo, Otero. Protocolo Auto-Organizado de Encaminamiento para Redes Submarinas No Uniformes (PAO-RSNU). VII Congreso Nacional de I+D en Defensa y Seguridad, 2019.
- Waheeduddin Hyder; Javier Poncela; Pablo Otero, Self-organized routing for radial underwater networks. 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 31 October 2016.
- Waheeduddin Hyder, Javier Poncela, Haji Khan Soomro, Shua Hussain, Comparison of MMSE and RLS Channel Estimation Methods for LTE-Advanced to Reduce Error in AMC Decision. ICBM, 28th – 29th of November 2015.