Improving Routing Performance of Underwater Wireless Sensor Networks

Beenish Ayaz

M.Sc. Information and Communication Engineering, Karlsruhe Institute of Technology, Germany

B.Sc. (Hons) Electronics Engineering, University of Engineering and Technology, Pakistan

A dissertation submitted in partial fulfilment

of the requirements for the degree of

Doctor of Philosophy

of the

University of Aberdeen

School of Engineering

June, 2016

Declaration

I hereby declare that this thesis has been written by myself and is based on the research work done by myself and that this thesis has not been presented for assessment in any previous application for a degree, diploma or other similar award. I also declare that all external sources of information have been specifically referenced.

Beenish Ayaz

School of Engineering

Fraser Noble Building, University of Aberdeen

Aberdeen, AB24 3UE

Acknowledgements

First of all I would like to express my profound gratefulness to almighty Allah. During this research journey, I had a chance to work with some extremely talented people. I would like to take this opportunity to express my heartfelt gratitude to them.

I want to express my sincere thanks to my supervisors Dr. Alastair R. Allen and Prof. Marian Wiercigroch for their constant encouragement, guidance and feedback in the fulfilment of this research work. Without their support and advice this research would have never been accomplished. Their outstanding mentoring throughout this research work is invaluable. I will always treasure this experience in my career. I am thankful to my fellow colleagues in Wireless Sensor Network Research Group and Electronics Research Group at University of Aberdeen. My sincere thanks to Dr. Nazila Fough, for being a great friend and support always. My special thanks to Dr. Golam Murshed, Dr. Celestine Iwendi, Chee Yun Yang, Dr. Raffaello Secchi, Dr. Ziaul Hossain and Eduard Grigorescu for their feedback and help at different stages of my research.

I am thankful to National Subsea Research Institute (NSRI), Scotland, UK for funding this research work. I would also like to extend my thanks to the anonymous reviewers for their comments that has greatly improved the content of the research papers from which this thesis has been partly extracted. My special thanks to the technical committee of judges of the 8th International Conference on Sensing Technology (ICST 2014) for bestowing me the “Best Presentation” award for presenting a part of this thesis work in a research paper. This award has proved to be a huge token of encouragement throughout this research work.

I am thankful to my wonderful husband Ayaz and lovely kids Emaan and Rayan for their sincere love, support, patience and sacrifice. And for always being there for me throughout this journey. Finally, I am deeply indebted to my great parents, Ikram ul Haq Ramay and Tabassum Ramay for their unconditional love, support and advice. For always encouraging me to pursue my dreams and giving me the vision to make the best out of life in every possible way. And special thanks to my lovely sister Mahvish for being an inspiration and my brothers Ali and Hasan for their care and encouragement.

Abstract

In this research work we propose a 3D node deployment strategy by carefully considering the unique characteristics of underwater acoustic communication as well as 3D dynamic nature of UWSN. This strategy targets 3D UWSN and not only improves the routing protocol performance significantly in terms of end to end delay and energy consumption but also provides reliability in data transmission. This strategy has been developed step by step from a single line of vertical communication to an effective 3D node deployment for UWSN. Several simulation experiments were carried out after adding different features to the final design to observe their impact on the overall routing performance. Finally, it is verified that this design strategy improves the routing performance, provides reliability to the network and increases network lifetime. Furthermore, we compared our results to the random node deployment in 3D, which is commonly used for analysing the performance of UWSN routing protocols. The comparison results verified our effective deployment design and showed that it provides almost 150% less end-to-end delay and almost 25% less energy consumption to the random deployment. It also revealed that by increasing the data traffic, our 3D node deployment strategy has no loss of data due to several back-up paths available, which is in contrast to random node deployment, where the packet loss occurs by increasing the data traffic. Improving the routing performance by carefully analysing the impact of 3D node deployment strategy and ensuring full sensing, transmission and back-up coverage in a highly unpredictable underwater environment, is a novel approach. Embedding this strategy with any networking protocol will improve its performance significantly.

Publications

- Ayaz, B., Allen, AR. & Wiercigroch, M., “Improving Routing Performance of Underwater Wireless Sensor Networks” in OCEANS '17 MTS/IEEE Conference, Aberdeen, UK, 19-22 June, 2017.
- Ayaz, B., Allen, AR. & Wiercigroch, M., "Dynamically Reconfigurable Routing Protocol Design for Underwater Wireless Sensor Networking" in Proceedings of the 8th International Conference on Sensing Technology, ICST 2014. Liverpool, UK, 2-4 September 2014.
- Ayaz, B., Allen, AR. & Wiercigroch, M., "Dynamically Reconfigurable Underwater Wireless Sensor Networking" in 5th Annual Engineering Postgraduate Research Symposium, Aberdeen, UK, 26 February 2014.
- Ayaz, B., Allen, AR. & Wiercigroch, M., "Dynamically Reconfigurable Routing for UWSN" in National Telford Institute Advanced Research Workshop, Aberdeen, UK, 23-24 Oct 2013

Achievement

- **"Best Presentation" Award** at the "8th International Conference on Sensing Technology (ICST 2014)", Liverpool, UK, 2-4 September 2014. http://www.abdn.ac.uk/news/6844/. Ayaz, B., Allen, AR. & Wiercigroch, M., "Dynamically Reconfigurable Routing Protocol Design for Underwater Wireless Sensor Networking"

Table of Contents

List of Figures

List of Tables

List of Acronyms

ACK	Acknowledgement
ADC	Analog to Digital Converter
ALOHA-AN	ALOHA with Advance Notification
ALOHA-CA	ALOHA with Collision Avoidance
AOSN	Autonomous Ocean Sampling Network
AUV	Autonomous Underwater Vehicle
CDMA	Code-Division Multiple Access
CMRE	Centre for Maritime Research and Experimentation
CSMA	Carrier-Sense Multiple Access
DBR	Depth Based Routing
DCB	Dynamic Cluster Based Routing
DESERT	DEsign Simulate Emulate Realize Test-bed
DFR	Directional Flooding based Routing
DUCS	Distributed Underwater Clustering Scheme
EM	Electromagnetic
EPROM	Erasable Programmable Read Only Memory
E2E	End-to-End
FAMA	Floor Acquisition Multiple Access
FBR	Focused Beam Routing

FDMA	Frequency-Division Multiple Access
FSO	Free-Space Optical
GPS	Global Positioning System
HH-VBF	Hop-by-Hop Vector-based Forwarding
H^2-DAB	Hop by Hop Dynamic Addressing Based
ILP	Integer Linear Program
LED	Light Emitting Diode
LM	Local Maxima
MAC	Medium Access Control
MBARI	Monterey Bay Aquarium Research Institute
MEMS	Micro-Electromechanical
MIMO	Multiple-Input-Multiple-Output
MVSA	Multipath Virtual Sink Architecture
NPS	Naval Postgraduate School
NS-MIRACLE	Network Simulator- Multi-InteRfAce Cross-Layer Extension
NS-2	Network Simulator - 2
NTF	Notification
NTU	Nephelometric Turbidity Units
OFDM	Orthogonal Frequency-Division Multiplexing
ONR	Office of Naval Research
OSI	Open System Interconnection
RAM	Random Access Memory

Rand	Random
REBAR	Reliable and Energy Balanced Routing Algorithm
RF	Radio Frequency
RISC	Reduced Instruction Set Computer
ROS	Robot Operating System
RRA	Resilient Routing Algorithm
SBR-DLP	Sector Based Routing with Destination Location Prediction
SDV	Small Delivery Vehicles
SEMAT	Smart Environmental Monitoring and Analysis Technology
SISO	Single-Input-Single-Output
SSC	Spawar Systems Centre
SUNSET	Sapienza University Networking framework for Simulation Emulation and Testing
TDMA	Time Division Multiple Access
TDS	Total Dissolved Solid
ToA	Time of Arrival
UAN	Underwater Acoustic Network
UUV	Unmanned Undersea Vehicles
UWSim	Underwater Simulation tool
UWSN	Underwater Wireless Sensor Network
UW-A	Underwater Acoustic Channel

VBF	Vector-Based Forwarding
WHOI	Woods Hole Oceanographic Institution
WOSS	World Ocean Simulator System
WSN	Wireless Sensor Network

CHAPTER 1

Introduction

Earth is a water planet, with about 70% of its surface covered with water, mainly in the form of oceans. If explored properly, oceans have a potential to contribute a substantial part in the betterment of life on earth. This could be in the form of energy supply for human and industrial needs. The oil and gas industry can advance further into underwater regions which have not been explored yet. The renewable energy can be harvested from the sea on a much larger scale. Similarly, tidal energy can be efficiently utilized for several applications. In addition to this, exploring and monitoring sea life, its habitat and minerals like cobalt, nickel, copper, silver, gold and rare minerals from the places which have not been reached before can be achievable. To realize all these goals, new offshore technology and infrastructure needs to be developed, maintained and repaired on a much wider scale than before. Considering the benefits of wireless sensor networks, scientists believe that underwater wireless sensor networks can play a vital role in achieving these goals in the underwater environment. Underwater wireless sensor networking is considered to be an extension of terrestrial wireless sensor networking, however, taking into account the wireless communication technology and propagation medium, the challenges experienced by Underwater Wireless Sensor Network (UWSN) are very different from terrestrial Wireless Sensor Network (WSN).

A typical UWSN consists of spatially distributed sensor nodes which are deployed in the underwater environment to perform collaborative sensing and monitoring tasks by forwarding the sensed data wirelessly from each other through the network to the destination node. The underwater environment is not a radio communication friendly environment like air. Radio frequency (RF) waves can only propagate through conductive salty water over long distance at extra low frequencies (30-300Hz) which requires large antennae and high transmission power. On the other hand, optical waves in underwater environment do not suffer from attenuation but are affected by scattering and require high precision in pointing the narrow laser beams [1] [2] . Hence the only practical means of wireless communication in underwater environment over reasonable distance is by means of acoustic waves. Due to high propagation delay, limited bandwidth capacity and high error probability of acoustic wave propagation underwater, it is extremely difficult to achieve the goals of terrestrial WSN in UWSN. Therefore, the benefits of WSN technology combined with acoustic wireless communication constraints in the underwater environment, establishes the basis for the challenging field of research, the UWSN technology.

1.1 Background

UWSN technology has gained profound interest in the past few years. Whether it is a natural disaster like a tsunami or an oil disaster, it is strongly believed in the research community that this can only be prevented by continuous real time monitoring with the help of UWSNs. Offshore exploration and sea life investigation are becoming increasingly challenging; areas which are not reachable by humans, need to be explored. UWSN communication makes it possible and is considered to be the future key asset in underwater monitoring, exploration and surveillance. Underwater monitoring systems used in the past were small-scale and used point-to-point single channel communication techniques such as remote telemetry and sequential local sensing etc. Real-time, highly precise and continuous spatio-temporal subsea data sensing systems are necessary for various applications, such as; subsea data collection, pollution monitoring, offshore exploration, disaster prevention, assisted navigation and tactical surveillance etc. UWSN technology possesses a high potential for research and development in every aspect. Despite the challenges, researchers and developers in this field are doing a remarkable work, however there is still a long way to go.

With the advancements in hardware technology, the design, development and improvement of routing protocols has always been a focus for research. A routing protocol ensures reliable data transmission from source node to the destination node. The performance of a sensor network highly depends on efficient forwarding of sensed data from source node to the destination node. This is a highly challenging goal to achieve in underwater environment. Sensor nodes communicate wirelessly, mainly by means of acoustic waves, hence there is high propagation delay, limited bandwidth capacity and high error probability as compared with radio wave communication. In addition to this, limited battery supply, three-dimensional (3D) underwater scenario and dynamic behaviour of sensor nodes due to water current makes the overall performance goals even difficult to achieve.

1.2 Motivations and Objectives

One of the key objectives of a sensor network is to provide a full sensing and communication coverage as well as connectivity with several back-up paths. However this goal is very difficult to achieve practically in an underwater environment due to the changing network configuration and challenging wireless acoustic communication. There are routing strategies which deal with the creation of void (a region created due to node movement or failure, which is not covered by any other node) by bypassing the void and re-routing the data around it towards the sink. In this way, any activity happening in the void region is completely ignored. In critical monitoring applications, transmitting data from selective regions can cause huge loss. Hence a system is needed which can ensure full sensing, communication as well as back-up coverage with uninterrupted connectivity for reliable data communication.

In this research work, an extensive literature review of the challenges faced by an UWSN was done and the focus was narrowed to ensure full sensing, communication and back-up coverage, hence providing reliability to the network communication and improving the routing performance . This is a challenging goal considering the 3D underwater environment. Mostly the routing protocol design concepts and node deployment strategies for UWSN are adopted from the terrestrial WSN, and re-designed with minor changes to fit in the underwater environment. Due to the fundamental differences between the UWSN and WSN technology, these designs are not suitable for 3D underwater environment. Although terrestrial WSN operate in 3D but the third dimension, which is the height is significantly small compared to the other two-dimensions (length and width of the network), hence the two-dimensional (2D) assumption for terrestrial WSN does not lead to any major inaccuracies. However a 3D network design, which is a prime requirement of UWSN, is much more complicated than 2D. For instance, the proofs of Kelvin's and Kepler's conjectures took centuries of research work to achieve a breakthrough, however their 2D counterparts are trivial to solve. Most of the networking protocols developed for UWSN are tested in 2D with random node deployment and the results are assumed to be acceptable to the actual 3D underwater environment. Random node deployment is simple to implement in the simulation tools. However keeping in mind the size, the sensing range and the communication range of UWSN nodes, it will cause major waste of resources. In addition with a random node deployment in 3D, there will be high probability of creation of void as well as congestion regions. Void is like a vacuum created in a sensor network with no coverage or connectivity. This is due to the absence of sensor nodes in that region, usually caused by node failure or movement due to water current. On the other hand, congestion regions are the regions covered by too many nodes and hamper the overall network operation due to congestion and repetitive task performance. It is evident that the results taken from random node deployment in 3D cannot guarantee the optimum use of resources to offer reliable communication, which can be offered to the network if the sensor nodes are properly deployed in a certain manner. Unlike tiny WSN nodes, UWSN nodes are big in size and with random deployment of such nodes, it is not feasible to have optimum use of these nodes for full coverage and connectivity. This was the motivation behind this research work.

The main objectives of this research work are summarised as below.

- To improve the routing performance of an UWSN in terms of end-to-end delay and energy consumption, by carefully considering the unique nature of underwater acoustic communication and specific characteristics of UWSN which makes it different from terrestrial WSN, such as 3D architecture and loss of connectivity due to node mobility etc.

- To ensure full 3D sensing, communication and back-up coverage as well as connectivity in a underwater environment in order to monitor all the activities in the sensed region.
- We aim to provide efficient and reliable data transfer in the region covered by the UWSN.
- We aim to increase the network life-time, by providing several back-up paths for communication. These backup paths will serve as extra resources to be used when needed.
- To ensure no loss of connectivity as well as data. Node mobility due to water current or node failure in underwater environment can cause loss of connectivity, several back-up paths will ensure no loss of data in this case.

1.3 Research Contributions

The contribution of our research work based on the aims and objectives is as follows.

- **Effective 3D node deployment strategy for UWSN.** In order to improve the routing protocol performance, an effective, yet simple 3D node deployment strategy has been proposed in this research work. It has been developed step by step from a single line of communication to an effective 3D node deployment strategy for underwater environment. Simulation experiment results were taken at each step in order to verify the effectiveness of introducing any feature to the design. Being analysed and tested through several simulations, it is verified that this design provides reliability to the UWSN at no extra cost to the resources.

- **Full sensing and transmission coverage.** Another main focus of our design is to ensure full sensing, transmission as well as back-up coverage. Basically, this is one of the prime objectives of a sensor network and somehow gets compromised due to the changing network configuration and challenging wireless acoustic communication. If a node goes missing and void is created, there are routing strategies which will still transmit the data by re-routing it around the void region. However any activity happening in the void region will be left as unmonitored and this is a violation of the prime purpose of a sensor network. A sensor network is installed to provide full coverage to the area it is monitoring. To transmit selective data or data from a selective area successfully is not what is required of a sensor network. With the 3D node deployment strategy we successfully address this issue by providing a full coverage to the sensed region.

- **Distribution of data load and energy consumption.** Our design ensures distribution of data load and energy consumption. In addition to main forwarding paths there are several back-up paths generated automatically in our 3D node deployment design at no extra cost to the resources. This does not only provide sensing coverage as well as transmission coverage in a 3D environment, it distributes the data and energy load. Due to the existence of several backup paths, certain regions will not drain batteries sooner than others. This is not possible to achieve with random node deployment. Even if random node deployment does not compromise significantly on overall energy consumption it can never ensure full coverage and connectivity and hence load distribution. Distribution of energy load resulting from our design will prolong the network life time.

- **Reliability.** Reliability is a prime focus of our design. With all the sensor nodes being well connected, having several backup paths, reliable sensing as well as transmission of data is ensured. We have introduced a term, Reliability Nodes in our design, due to their sole purpose of providing reliability and increasing the network lifetime without causing any extra resource consumption. Even by increasing the data traffic, our 3D node deployment strategy ensures reliable transmission with no loss of data due to several back-up paths available which is in contrast to random deployment where the packet loss occurs by increasing the data traffic.

- **Remarkable decrease in end-to-end delay.** As a part of routing protocol performance improvement, we achieved a significant decrease in end-to-end delay from our effective 3D node deployment strategy. This is particularly important in UWSN, where due to acoustic communication, the speed of transmission of data is 5 orders of magnitude less than in radio communication. In addition to this, the sensor networks always strive for real-time data transmission. Hence this is a remarkable achievement. Our 3D node deployment design improves the transmission time significantly by decreasing the end-to-end delay for the transmission of data from source node to the destination node.

1.4 Thesis Organization

This thesis is structured in 7 chapters.

Chapter 2. In Chapter 2, the background knowledge to understand the UWSN technology is presented. The communication architecture of an UWSN explaining its functionality and the components that constitute a sensor node are discussed. Then this chapter provides the scope to understand the distinctive features of an UWSN that makes it entirely different from terrestrial WSN. Further in this chapter the wide range of UWSN applications are highlighted. In the end the challenges faced by an UWSN are presented.

Chapter 3. In Chapter 3 underwater wireless communication is discussed. The main physical layer technologies (acoustic, electromagnetic and optical) used for wireless communication in underwater environment are investigated. Several fundamental key aspects of wireless acoustic communication in underwater environment are highlighted and it is discussed why acoustic propagation is the preferred mode of wireless communication in UWSN. After the fundamental concepts of underwater communication, UWSN protocol stack is discussed. And finally an overview of simulators and testbeds available for experimenting an UWSN are presented in the end.

Chapter 4. In Chapter 4 the routing algorithms for UWSN are investigated in detail highlighting the main issues in order to suggest improvement techniques. One of the major common flaws is that the performance evaluation of these protocols is done either in 2D or in 3D by random node deployment. Underwater environment is 3D, 2D node deployment is not applicable and 2D results cannot be interpreted for 3D. Random node deployment in 3D cannot ensure reliable transmission of data with optimum use of resources.

Chapter 5. In Chapter 5 an effective 3D node deployment strategy for UWSN is proposed in order to improve the routing protocol performance. It is developed step by step from a single line of communication to an effective 3D node deployment. The impact of Reliability Nodes, a term introduced in our design for nodes providing reliability, improving the routing performance and increasing the lifetime of a network, is discussed. This strategy is analysed and tested through several simulation experiments. In addition, it is verified that this design provides reliability to the UWSN at no extra cost to the resources. It ensures full sensing, transmission and back-up routes coverage to the UWSN. The overall energy consumption and end-to-end delay of the network is improved significantly. As well as improved data load and energy consumption distribution is achieved by adding the Reliability nodes to the 3D cubic lattice node deployment strategy. These nodes cost no extra energy to the network hence energy cost is distributed and network lifetime is increased.

Chapter 6. In Chapter 6, we analyse the performance of our 3D node deployment strategy presented in Chapter 5, against several test set-ups. The total energy consumption, energy consumption per node, total end-to-end delay and delay per node for all the test set-ups are analysed and discussed. This is done in comparison with Test Rand, which is the commonly used random 3D node deployment strategy, for testing the performance of any networking protocol. It is revealed that this approach has significant drawbacks, which will affect the overall network performance adversely, no matter how efficient the networking protocols are designed to perform. Further we show that our proposed 3D node deployment strategy which is simple to implement yet effective, performs much better than most of the 3D test set-ups and even better than most of the 2D test set-ups in absolute terms on per node basis.

Chapter 7. Finally Chapter 7 concludes the thesis and presents the overall conclusion and original contributions of this research work. Further it outlines the prospective topics for future research in this direction.

CHAPTER 2

Underwater Wireless Sensor Networks

In this chapter, the background knowledge to understand the UWSN technology is presented. A basic understanding of the UWSN architecture and its functionality is discussed, highlighting the components that constitute a sensor node, which is the main building block of a sensor network. Then this chapter provides the scope to understand the distinct features of an UWSN that makes it entirely different from a terrestrial WSN. Further in this chapter the wide range of UWSN applications are presented. Finally, the challenges faced by an UWSN due to the very nature of wireless networking in underwater environment are discussed.

2.1 Communication Architecture of an UWSN

UWSN consists of a number of spatially distributed underwater sensor nodes which are wirelessly connected to each other and distributed in a 3D underwater volume (which is intended to be monitored). The prime purpose of these sensor nodes is to provide full sensing as well as transmission coverage to the observed region. In this way, any activity happening in the observed region (covered by the sensor network) can be sensed immediately and the information relating to that activity can be forwarded wirelessly to the base station with the help of other nodes. Basically the sensor nodes are deployed in a wireless network to perform collaborative monitoring tasks by relaying the sensed data from one another through the network to a data sink and further to the base station. Autonomous underwater vehicles (AUV) can also aid the UWSN operation if the region observed is very large. A typical communication architecture of an offshore oil field comprising of several UWSNs is shown in Figure 2.1. There are some powerful anchored beacon nodes near the seabed. These beacon nodes may act as a backbone to the monitoring activities, providing localization or acting as gateway nodes for some applications, depending on their location.

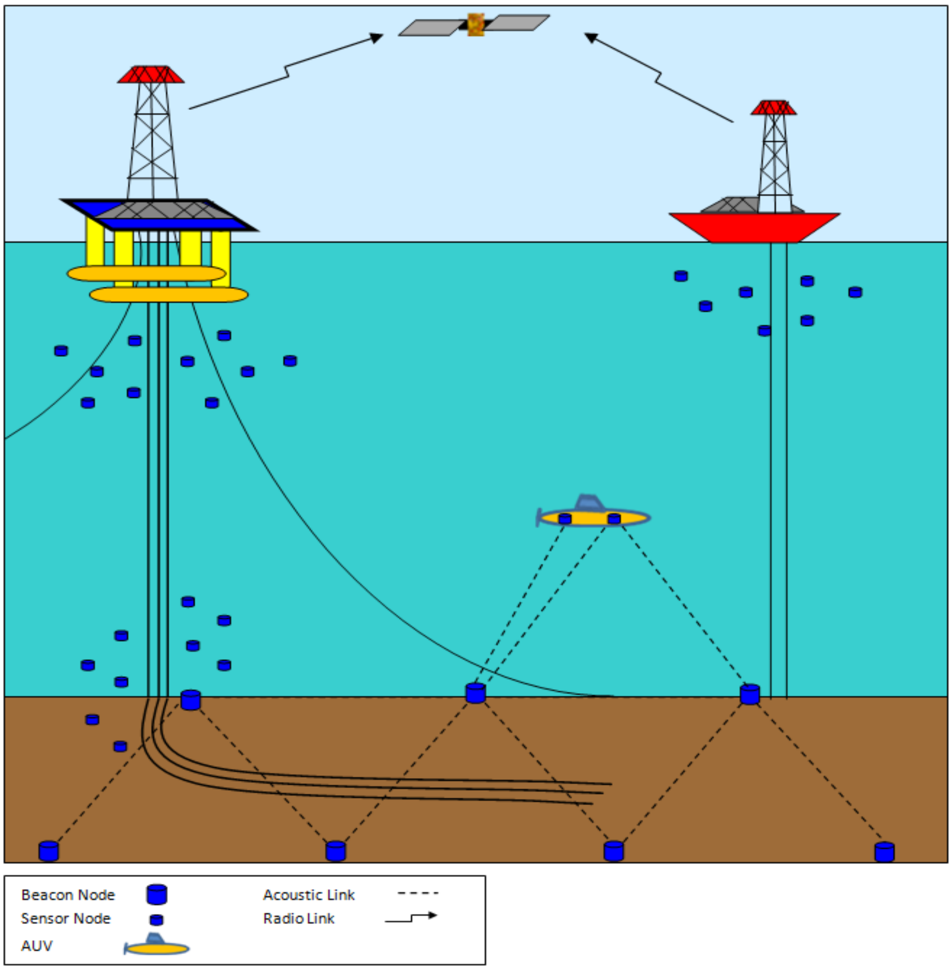


Figure 2.1: Typical communication architecture of an offshore oil field comprising of several UWSNs

The wireless link between the nodes is acoustic, as acoustic waves are the preferred mode of communication in the underwater environment. When the data reaches the surface, it can be transmitted in the form of electromagnetic waves. After achieving full coverage with the sensing and communication range of the sensor nodes in an UWSN, the number of sensor nodes can be increased near the critical monitoring areas to increase the overall life time of the network and reliability. In a large UWSN, the number of sensor nodes can vary from place to place and some less active areas can be covered by AUVs to relay data as shown in the Figure 2.1.

2.2 Underwater Wireless Sensor Node

The main component of an UWSN is a sensor node, which can be thought of as a resource constrained basic computer with some sensing capability. It possesses limited memory and processing ability and can sense, process, store and transmit data. It can also be referred to as a transducer (based device) that converts the signal from one form of energy to another. For example it can sense light, salinity, pressure, temperature, density, chemical components, conductivity, PH or turbidity etc. in water and can transmit the sensed data in the form of a wireless acoustic signal to another sensor node, where this data can be interpreted, processed and forwarded accordingly. A sensor node can be acoustic, thermal, optical, chemical etc., depending on the physical quantity it can measure or sense and transmit.

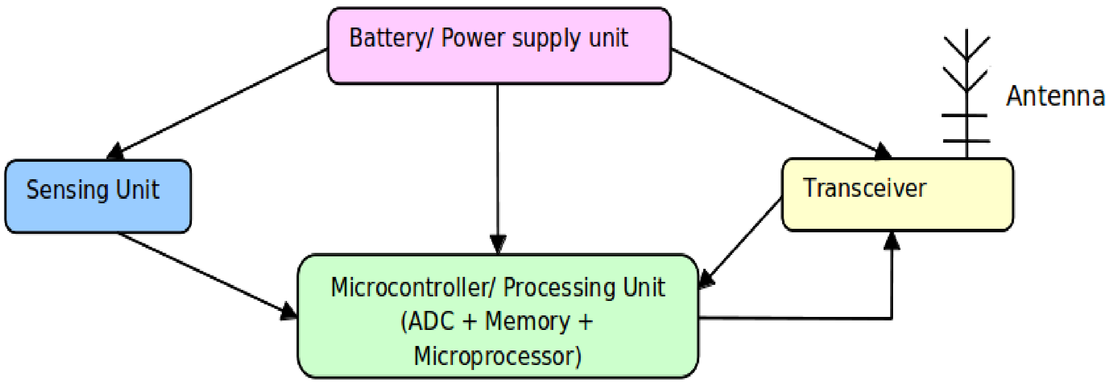


Figure 2.2: Basic Sensor Node Architecture

An UWSN node possesses similar technical architecture to a wireless sensor network (WSN) node. However, it requires more power to support acoustic communication, more memory for data caching in order to keep up with the high error probability of an acoustic link, is bigger in size and is enclosed in a protective housing to avoid corrosion. As shown in Figure 2.2, the basic components of a sensor node are,

- sensing unit (a micro-electromechanical (MEMS) sensing device and analog to digital converter (ADC))
- processing unit (a microcontroller with limited memory)
- power supply unit (a battery)
- transceiver (acoustic, optical or radio)

2.2.1 Sensing Unit

It consists of a micro-electromechanical (MEMS) sensing device and an analog to digital converter (ADC). This unit interacts with the surroundings to sense a change in the physical quantity it is required to measure. Then it converts that analog signal to a digital signal for further processing by the processing unit.

2.2.2 Processing Unit

This unit consists of a microcontroller with limited memory and is designed to operate with low power. This component performs local processing of the sensed data within the sensor node and local decision making depending on the sensed data. It also performs the local tasks like networking and choosing the best forwarding node etc. These microcontrollers are usually 16 or 32-bit Reduced Instruction Set Computer (RISC) architecture based, and run at low frequencies. They save power by operating in different energy saving modes, such as sleep mode and active mode. When there is no activity or change in the sensed data they stay in sleep mode and do not send any data signal to the transceiver. However if the sensing unit senses any new data the processing unit wakes up from sleep mode into active mode, processes the data and sends it further through the transceiver to the next forwarding node. The processing unit possesses on-chip Random Access Memory (RAM) for the storage and programming of data. This memory is usually a few hundreds of kilobytes. In addition there can be some Erasable Programmable Read Only Memory (EPROM) and Flash memory.

2.2.3 Power Supply Unit

The power supply in a sensor node is limited and is in the form of batteries. There are mainly two kinds of batteries: Alkaline and Lithium. Alkaline batteries are larger in size, cheap and high capacity but variable voltage supply which is not acceptable for some sensors. Lithium batteries are much smaller in size compared to alkaline batteries and have constant voltage supply at temperatures as low as -40 ^{0} C even when the battery is almost drained. Optimizing and selecting a suitable power supply unit for underwater sensor node is an important factor as it will directly affect the efficiency and life time of an UWSN. During processing and networking tasks, saving energy is of top priority. There are a lot of energy scavenging approaches being explored for underwater sensor nodes. For example, difference of temperatures or pressures or the water movement in the sea can be utilized to produce small amounts of energy sufficient enough to perform local processing and transmitting a data signal.

2.2.4 Transceiver

This unit is responsible for sending and receiving a data signal wirelessly to and from other nodes. Underwater transceiver can be acoustic, optical or radio. It usually comprises of the following components: a modulator, demodulator, digital to analogue and analogue to digital converter, filters, low noise and power amplifiers, mixers and antenna. To save power, transceivers operate in different modes such as transmit, receive, idle and sleep mode. Transmission power, modulation schemes and data rates vary in different transceivers.

2.3 Distinctive Features of UWSN

An UWSN is conceptually similar to terrestrial WSN, and it shares some of the common features of terrestrial WSN. However certain distinctive features make it entirely different technically from a terrestrial WSN. The high impact of these differences makes the networking protocols developed for WSNs not suitable for UWSNs. Some of the distinct features of an UWSN are discussed in the following sections.

2.3.1 Mode of Communication

The main mode of wireless communication in an UWSN is acoustic waves unlike radio waves in terrestrial WSN. Other physical layer technologies like radio [3] and optical [4] [5] are often impractical in the underwater environment due to their relatively high absorption or scattering, and have only been proposed for short-range links (typically 1-10m) where their very high bandwidth (MHz or more) can be exploited. In fact, radio waves propagate through conductive salty water only at extra low frequencies (30− 300 Hz), which require large antennae and high transmission power. Optical waves do not suffer from such high attenuation but are affected by scattering. The transmission of optical signals requires high precision to point the narrow laser beams. Acoustic communication offers longer range but imposes limitations on the overall network design due to slow propagation speed, limited bandwidth capacity and high error probability of acoustic waves.

2.3.2 Slow Propagation Speed

The propagation speed of acoustic waves is 1500 m/s in water, which is 5 orders of magnitude lower than electromagnetic waves. This much slower speed of acoustic propagation (compared with electromagnetic and optical waves), low bandwidth capacity and high error probability are the limiting factors for efficient communication and networking [6], compared with WSN. It is not possible to achieve the terrestrial WSN communication capabilities and networking goals in UWSN. The slow propagation speed of acoustic waves have a serious effect on time synchronization, medium access control protocols, reliable end-to-end real-time communication and dynamic reconfiguration strategies. Therefore the networking protocols used for WSN cannot be applicable to UWSN unless they are carefully modified for the acoustic underwater communication parameters.

2.3.3 High Error Probability

The acoustic link in an UWSN suffers from high bit error rates. This is caused due to multiple factors, like absorption loss, path loss, noise, multi-path, Doppler spread and high propagation delay etc. These factors are discussed in chapter 3. This makes a wireless acoustic link very different from a radio link.

2.3.4 Low Bandwidth Capacity

The underwater acoustic channel has limited bandwidth, in the order of kHz depending on the range and frequency. For a long-range system, operating over 10-100 km, the bandwidth is limited to a few kHz, for a very long distance in the order of 1000 km, the available bandwidth falls below a kHz, a medium-range system operating over 1-10 km has a bandwidth in the order of 10 kHz, while only for very short ranges below about 100m, more than a hundred kHz of bandwidth may be available [1] . This implies that the long range acoustic communication link (~tens of km) only has a bandwidth of a few kHz. A short range link (~100 m) may have a higher bandwidth (>100 kHz) but in both cases these factors lead to low bit rates [7]. Table 2.1 [2] shows the available bandwidth for different ranges in underwater acoustic channel (UW-A).

Table 2.1: Available Bandwidth for Different Ranges in UW-A Channels [2]

	Range (*km*)	**Bandwidth (*kHz*)**
Very long	1000	<1
Long	10-100	2-5
Medium	1-10	~10
Short	0.1-1	20-50
Very short	<0.1	>100

2.3.5 Dynamic Network Configuration

Most ground based WSN nodes are static, however UWSN nodes are often mobile due to water current, unless they are anchored to the seabed or mounted on underwater equipment for special applications. Due to water current the network configuration keeps on changing. Empirical observations suggest that water current moves at a speed of 3-6 km/h in typical underwater conditions [8] . In addition to this, node failure, battery failure and channel impairment can cause the network configuration to be changed. Therefore to ensure full sensing and transmission coverage in an UWSN, overcoming the impact of changing network configuration is a challenging issue to deal with.

2.3.6 Three Dimensional (3D) Network Configuration

Another distinguishing feature of an UWSN is its 3D configuration. Although a terrestrial WSN also operates in 3D, the third dimension, which is the height, is significantly small compared to the other two dimensions (length and width of the network), hence the two-dimensional (2D) assumption for terrestrial WSN does not lead to any major inaccuracies. In UWSN the third dimension is as significant as the other two dimensions. Therefore, the 3D network design is of prime importance in UWSN and it is far more complicated than 2D network design.

2.3.7 Cost

Underwater sensor nodes are not only bigger in size but are expensive compared to WSN nodes. In addition to this the deployment as well as maintenance techniques used for these nodes are also costly. Similarly the UWSN research experiments on real hardware in the sea are expensive to perform and mostly the research community rely on simulators and experimental testbeds for this purpose.

2.3.8 Lack of Common Experimenting Platform

Underwater wireless sensor networking research experiments in the field on the real hardware are very expensive. To facilitate researchers, a number of simulators have been developed, but unlike terrestrial WSNs there is no common reliable platform for experimenting. There are a number simulators developed, almost as many as the number of networking protocols. However, most of them lack the complexity of physical layer hence the simulation results can differ from the real in-field results significantly. An intermediate solution is testbeds, some of which have recently been made available to other researchers. There are also some frameworks which claim to support the user all the way from simulation to hardware testing, however proper documentation and user manuals of these tools are still not available. Due to this the researchers have to struggle to experiment their techniques.

2.4 Applications of Underwater Wireless Sensor Network

There is a wide range of applications of UWSNs in the challenging underwater environment. The tremendous amount of software and hardware research going on in the field of UWSN, is making these applications possible. UWSN are envisioned to play a vital role in the next generation of subsea monitoring and exploration systems. Some of the main applications are as below.

2.4.1 Subsea Data Sampling

A network of sensors and AUVs, wirelessly connected to each other can perform collaborative adaptive sampling of the 3D subsea environment. An Autonomous Ocean Sampling Network (AOSN) project at the Monterey Bay Aquarium Research Institute (MBARI) demonstrates the advantages of bringing together sophisticated new robotic vehicles with advanced ocean models to improve the ability to observe and predict the characteristics of the oceanic environment [9] [10] .

2.4.2 Pollution Monitoring

Water is an important natural resource for the survival of life over and underwater. Pollution monitoring in underwater environment is a very important application of UWSNs. To monitor the chemical constituency of seawater, pH (potential of hydrogen, is a numeric scale used to specify the acidity or basicity of an aqueous solution) levels and any harmful chemicals that can harm sealife or the life over sea is possible with the help of several wirelessly communicating sensor nodes. These will sense the presence of any harmful chemicals or intolerable imbalance in the chemical constituency of seawater which can adversely affect the marine life. Further samples for detailed investigation can be taken with the help of AUVs. In [11], the authors have presented a decentralized UWSN for ocean pollution prevention. It is a short range (50-500m), low cost sensor node based design, which focuses on short range communication avoiding the challenges of long range underwater communication, hence greatly simplifying the modem design.

2.4.3 Offshore Exploration

There are large amounts of natural resources (like oil and gas) and precious minerals under the seabed. Exploring the oil and gas reserves underwater is an important application of UWSNs, as it can benefit from wireless communication and can avoid expensive cable network on relatively large areas. A self configuring UWSN can be set up at certain locations to explore oil fields. A large scale UWSN to explore oceanic environment and mineral resources underwater is proposed in [11] . The sensor nodes are installed with video capturing devices to investigate the underwater resources, and real-time video streaming is proposed.

A network protocol ACMENet [12] is proposed for deep sea exploration and monitoring. It is claimed to be efficient in terms of underwater communication bandwidth and sensor node battery consumption. The protocol is tested in sea trials within the framework of the ACME project sponsored by the European Union. Further literature on developing an UWSN for deep sea exploration is presented in [13] .

2.4.4 Disaster Prevention

Another important driving force behind the research and development of UWSNs is disaster prevention applications. Natural disasters and in particular water based natural disasters cause huge destruction. In addition, oil spill disaster can result in significant instabilities in the above and underwater ecosystems. UWSNs that monitor seismic activity remotely can provide Tsunami warnings. By monitoring and co-relating the events and occurrences that cause such a disaster, an early warning can save millions of lives.

In [14] the physical layer challenges in establishing a reliable, low power consuming UWSN system for early warning generation in case of any natural disasters like tsunami is discussed. Similarly a sense and response system is proposed in [15] for tsunami detection and mitigation. Further literature on this can be found in [16] , highlighting underwater natural disaster monitoring applications. However, very few UWSNs for seismic monitoring and Tsunami detection are practically deployed, most of the strategies have only been simulated and tested on testbeds. A mechanical design and hardware implementation of a sensor node that is capable of sensing, processing, and transmitting information about an oil spill (location and thickness) is presented in [17] . Also two new methodologies of detection, based on the difference in the absorbance spectral signatures and electric conductivity properties of oil and water are proposed. The experimental results can be plotted on a map to show the location.

2.4.5 Marine Life Monitoring

Monitoring marine life with the help of UWSNs is another challenging application. This involves fish farm monitoring, other habitat monitoring like micro-organisms, plants and coral reefs. There has been a lot of marine life monitoring applications developed to date like [18] [19] [20] [21], but most of them utilize wireless sensor radio communication and can only be practical for shallow water application. Marine life and habitat exists mostly deep underwater. UWSN existence came into being mainly for monitoring the areas which cannot easily be accessed by human. Hence the use of UWSN which are acoustic communication based will provide a the reliable means of properly monitoring marine life in larger regions underwater. This claim is supported and verified in [22], where the authors have presented the lessons learned from two years of operation of wireless sensor networks deployed at seven coral reefs along the Great Barrier Reef in north-eastern Australia and have clearly highlighted the impractical approach of using terrestrial wireless sensor networks for underwater applications. They finally propose a practical, real-time, acoustic communication based UWSN for monitoring of coral reefs. The ACMENet [12], a European funded project proposed for deep sea exploration and monitoring, is also capable of underwater marine life monitoring. Similarly, Smart Environmental Monitoring and Analysis Technology (SEMAT) [23] is a cost effective marine and coastal environmental monitoring system.

In [24] a detailed review of the impact of anthropogenic noise on marine life is discussed, there is no clear evidence of any impact due to the frequency range (~kHz) used by acoustic sensor network technology. It is also revealed that only some aspects and an extremely small portion of marine organisms have been investigated and this study is limited. However the impact of anthropogenic ocean sound from shipping has been monitored using acoustic sensor network in [25].

2.4.6 Subsea Equipment Monitoring

UWSNs can play a vital role in efficiently monitoring the subsea equipment after installation for any failures or problems. There is a large network of oil and gas pipelines under the sea which need continuous monitoring for any breakdown, blockage or leak and UWSNs are very appropriate for this application. A detailed survey of the network architectures for monitoring the pipelines is presented in [26] and an underwater integrated wired/acoustic wireless sensor network is proposed. In this network, the power is provided to the nodes through wired connection and the communication is through wireless acoustic links. Hence the sensor node batteries will be used only if there is a fault in the wires providing power to the nodes. As a result, the network life time will be significantly increased as compared with the ordinary acoustic wireless sensor network.

It can be summarized that only shallow water quality monitoring and oil spills near the sea surface can be monitored using wireless sensor networks utilizing radio frequency links. This is due to the fact that radio frequency communication in underwater environment is highly restricted with distance. For longer distance deep sea monitoring only acoustic communication is practical.

2.4.7 Assisted Navigation

Underwater environment is highly unpredictable, uneven and dark. Navigation of vessels, ships, submarines or even divers can be very challenging. UWSNs can play an important role in providing localization and navigation facility, even to regions which are not easily accessible. Sensor nodes can also identify hazards or mines underwater and can assist in bathymetry of the sea, providing useful data for other applications. Terrestrial navigation facilities cannot be utilized underwater as the mode of communication will be acoustic to cover longer distances. A highly efficient GPS (Global positioning system) for underwater application is needed. In [27] a detailed overview of possible UWSN localization schemes is discussed. A few scenarios have been considered, however, all the nodes involved in localization are stationary. An AUV-aided UWSN for localization is presented in [28] and it is shown that the higher communication cost can be minimized while maintaining good performance with an appropriate choice of the acoustic communication range. The sensor nodes deployed in this scheme are stationary and the performance is tested on the basis of localization coverage, accuracy and communication costs. An anchor-free localization algorithm for underwater wireless sensor network is proposed in [29] , in which the relationship of adjacent nodes is used for designing the algorithm, however while testing the localization scheme the underwater complex communication environment has not been considered.

In [30] the authors have presented an on-demand underwater localization algorithm for locating an arbitrary number of nodes via asynchronous anchored nodes. This scheme takes advantage of a sequential transmission protocol considering the broadcasting nature of the underwater acoustic medium and can locate both initiating and passive nodes within the network with a small overhead. The algorithm has been implemented on an existing modem and has been tested within a pool and lake.

2.4.8 Distributed Tactical Surveillance

UWSNs can be used for tactical surveillance, reconnaissance, targeting and detecting intruders. This can be helpful in military and naval operations. An adaptive autonomous underwater vehicle for littoral surveillance have been presented and tested in the GLINT10 field trial [31] . The on-board signal processing capabilities and the implications for underwater surveillance using AUVs are discussed. The implementation and at-sea testing of an underwater acoustic network, an EU-funded project (UAN11) is presented in [32]. It is composed of fixed as well as mobile nodes. The mobile nodes are routinely added and removed from the network with the help of AUV's, hence a complicated underwater acoustic network has been presented and tested. The at-sea testing results gathered continuously over five days showed robustness of the network to tackle variation in its structure (with nodes routinely entering and leaving the network) and poor communication performance, with large and variable delays and packet loss. However, the recorded data showed that the security overhead did not degrade the overall network performance any further. An investigation involving better communication performance is needed to verify this conclusion.

A distributed 3D space coverage scheme for tactical underwater sensor networks is proposed in [33] where sensor nodes transmit their data packets through cables to surface buoys. In this scheme an appropriate depth for each node is determined so that the maximum 3D coverage is maintained. Each node is equipped with multiple micro-sensors such as acoustic, magnetic, radiation and mechanical. The readings from these sensors help detecting and classifying the object e.g. submarines, small delivery vehicles (SDVs) and divers. In comparison to traditional sonar systems, underwater sensor networks can provide higher accuracy in detecting the low signature targets by combining the sensed data from different types of sensors. In [34] a big project is presented, comprising of fibre optic cable linked, large-scale ocean bottom observatory network, for tsunami monitoring along the Japan Trench. This network will utilize real-time data to estimate the scale and extent of tsunami warning.

2.5 Challenges faced by an UWSN

UWSN faces some challenges similar to terrestrial WSN which are due to the very nature of wireless (underwater) sensor networking technology. In addition it faces more issues due to the mode of communication which is acoustic in UWSN. The challenges and issues of a UWSN are discussed below.

2.5.1 Power Management

Typically the power source is limited (battery). These limited batteries along with the networking protocol management and environment define the life of a sensor network. The power required for UWSN communication is higher than the terrestrial WSN communication due to the difference in physical layer technologies (mainly acoustic for UWSN and radio waves for WSN). In addition, the larger distances and complex signal processing techniques used to compensate for the acoustic channel impairment at the receivers end, causes more energy overhead. Effective power management or some means of energy harvesting can extend the battery life. For example, difference of temperatures or pressures or the water movement in the sea can be utilized to produce small amounts of energy. Similarly solar energy can be utilized to charge batteries, this will involve some robotics to push the drained rechargeable batteries to the surface charging station and replace them with recharged batteries. Autonomous underwater vehicles (AUV) can aid this process.

2.5.2 Limited Memory and Processing Capability

The sensor nodes possess limited memory and processing capability as the power supply is limited. While designing an UWSN, any complicated computation which requires high data processing, memory and energy resource, needs to be avoided in order to increase the life of the network. The sensor node memory and processing unit is designed to perform local tasks, like processing their own data so that it can be forwarded to the neighbouring sensor nodes successfully and keeping a count of the forwarded data packets in the memory to avoid duplicate transmission.

2.5.3 Node Failure/ Broken Links

This normally happens when a node dies (due to running out of battery or other design failure) in a static WSN. In UWSN the node mobility due to water current and channel impairment also plays a role [35]: an underwater sensor node can move out of line of communication or can enter line of communication due to water current. To over come the former possibility a dynamic reconfiguration process is proposed in [35] to re-establish the communication links if the network configuration changes. When a node goes missing and void is created, there are routing strategies which can still transmit the data by re-routing it around the void region. However any activity happening in the void region will be left as unmonitored and this is a violation of the prime purpose of a sensor network. A sensor network is expected to provide full coverage to the area it is monitoring. A proper 3D node deployment strategy can ensure sensing as well as transmission coverage by providing several back-up paths.

2.5.4 Proper Wake up/ Sleep Schedules

A sensor node is designed to wake-up for certain period of time to communicate, when it senses an activity and if there is no further activity sensed after sometime, then it goes back to sleep mode. This is highly effective way of saving energy in UWSN as well as WSN. However, designing a proper wake/sleep mode can be challenging so as not to affect the synchronized data transmission involving a number of sensor nodes. This has to be taken care effectively by MAC (Medium Access Control) layer, in order to save energy, avoid congestion and utilize the time slots efficiently by clock synchronization.

2.5.5 Reliable End-to End Communication

This is usually expected from the transport layer but in case of UWSN this will be taken care by MAC or routing layer at each hop due to the changing network configuration. Especially when the sensor nodes are wirelessly connected they can move due to water current, hence taking into account their mobility a networking protocol should be able to predict their position for reliable transmission. Similarly reliable coverage can be ensured by means of back-up routes.

2.5.6 Congestion Control

In UWSN as well as WSN there can be high congestion areas compared with others, depending on the location of the monitored activity. If two or more activities are sensed simultaneously, then congestion is an issue to deal with to avoid bottle necks around the data sink. In UWSN, limited bandwidth makes the congestion issue more complex to deal with which makes the protocol design even more challenging. One possibility can be using clusters with powerful cluster heads which after collecting the data relay it to only other powerful nodes/cluster heads only. Another possibility can be using multi-sink architecture and directional communication as proposed in [35] .

2.5.7 Sensor Localization

It is usually required to know the exact location of the event hence beacon nodes (nodes that know their location) need to be used efficiently. GPS can be one option but it will make the whole network design complicated and costly. To avoid this, there are two popular localization techniques, as stated below [7] .

i. *Range based* : First determine the range/distance between the nodes then apply geometric principles to calculate the exact location.
ii. *Range free scheme* : In this case, from hop counts one can estimate distance between nodes using an average distance per hop. Range free is not as accurate as range based but does not require extra hardware and computation by the sensor nodes to determine distance.

Depending on the application and criticality of the monitored activity this can be designed efficiently.

2.5.8 Scalability

With everyday passing, underwater communication is advancing involving more and more automation and control. In order to achieve this and to extend the existing underwater networks, a scalable design is important so that one can add and integrate more features along the way. A robust network design should be able to expand to accommodate more functionality.

2.5.9 Security

This is another important area to be taken care of in an UWSN design, in order to protect the network from intruders and secure the data transmission. There are several security techniques as discussed in [36] [37] . The use of chaotic signals for underwater wireless communication allows a noise-robust secure transmission which blends naturally into the background noise and thus can be difficult to detect by an intruder.

2.5.10 Node Density

In an UWSN design, node density plays an important role. Increasing the node density will increase the network reliability as well as cost and congestion. An optimum trade-off needs to be considered in order to save cost. The basic purpose of a sensor network is to provide sensing, transmission as well as back-up coverage to the region it is monitoring. Increasing the number of nodes more than the optimum number to provide full coverage needs to be done carefully. This will involve careful consideration of unique nature of underwater acoustic communication such as high and variable propagation delay, low bandwidth capacity, high error probability and temporary loss of connectivity as well as specific characteristics such as 3D architecture and loss of connectivity due to node mobility.

2.6 Summary

This chapter presents the background knowledge to understand the nature of the field of UWSN. The architecture of a sensor network and its main component i.e. a sensor node, is described. The distinct characteristics of UWSN are discussed which outlines the difference between a terrestrial WSN and an UWSN technology. These include mode of communication, slow propagation speed, limited bandwidth capacity, high error probability, dynamic and 3D network configuration and cost. The wide range of application areas of an UWSN are covered such as, subsea data sampling, pollution monitoring, offshore exploration, disaster prevention, marine life and subsea equipment monitoring, assisted navigation and tactical surveillance etc. Finally, the challenges faced by an UWSN are highlighted which are of prime interest for the research community.

CHAPTER 3

Underwater Wireless Communication

This chapter describes underwater wireless communication techniques. The main physical layer technologies (i.e. acoustic, electromagnetic and optical) used for wireless communication in underwater environment are investigated. Several fundamental key characteristics of wireless acoustic propagation in underwater environment are highlighted and it is discussed why acoustic propagation is the preferred mode of wireless communication in UWSN. Then the protocol stack for UWSN is discussed. And finally an overview of simulators and testbeds available for experimenting an UWSN are presented.

3.1 Acoustic Communication

Acoustic communication is the main physical layer technology of an UWSN due to the low absorption of sound waves in the underwater environment. This is especially true in thermally stable deep water. On the other hand, the use of acoustic waves in shallow water can be adversely affected by temperature gradients, surface ambient noise, and multipath propagation due to reflection and refraction. The main features of acoustic communication are as follows.

3.1.1 Propagation Speed

The propagation speed of an acoustic signal in water is directly proportional to water temperature, salinity and pressure (which is directly related to underwater depth) [6]. A typical speed of sound in water is about 1.5 x 10^{3} m/s, which is 4 times faster than in air and 5 orders of magnitude lower than the speed of light (3 x 10^{8} m/s) [6]. The sound speed including the effect of both temperature and pressure as a function of depth is shown in Figure 3.1 [6], the variation in sound speed is non linear as in general. However for rough quantitative and qualitative discussions, for 1^{0} C rise in temperature the speed of sound increases by approximately 4m/s, for 1 practical salinity unit (PSU) rise in salinity, the speed of sound in water increases by approximately 1.4m/s, for 1km increase in depth the speed of sound increases roughly by 17m/s [6].

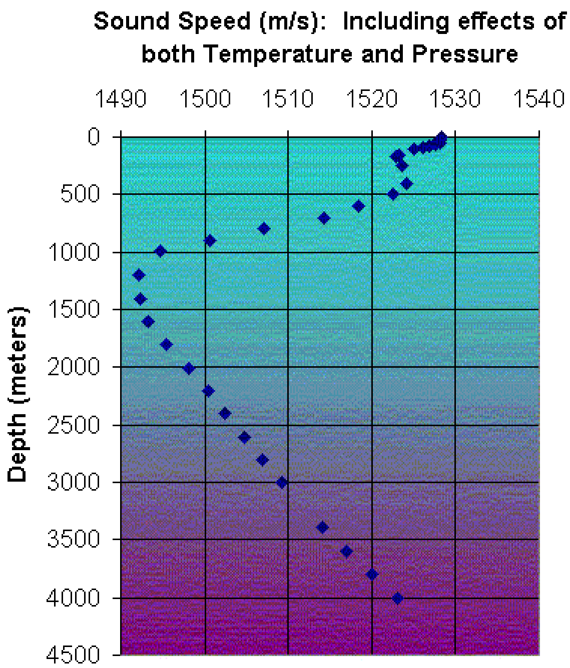


Figure 3.1: Sound speed in sea water as a function of depth [6]

During underwater propagation an acoustic signal incurs path loss (signal attenuation), which is mainly caused by geometric *spreading loss*, *absorption loss* and *scattering loss* .

3.1.2 Path Loss

In an acoustic channel, path loss is mainly due to the combined effect of geometric spreading loss, absorption loss and scattering loss and it depends on the signal frequency and the transmission distance. It can be given by equation 3.1 [38],

$$A(d,f)=\left(\frac{d}{d_r}\right)^k \alpha(f)^{d-d_r} \quad , \tag{3.1}$$

where *f* is the signal frequency and *d* is the transmission distance, taken in reference to some distance d_r , exponent *k* models spreading loss and its usual values are between 1 and 2 (for cylindrical and spherical spreading, respectively) and *α(f)* is absorption coefficient at frequency *f* [6]. *f* increases, the absorption coefficient *α(f)* increases, hence the high frequency waves will be considerably attenuated within a short distance, while low frequency acoustic waves can travel very far [6]. As a result, the bandwidth is extremely limited for long-range applications, while for short-range applications, several tens of kHz bandwidth could be available [6].

3.1.3 Geometric Spreading Loss

The propagation of an acoustic wave from a sound source spreads the transmitted acoustic energy over a larger and larger surface. As energy is conserved, the intensity will decrease proportionally to the inverse of the surface, this process is known as geometric spreading loss which increases with distance as illustrated in Figure 3.2 [39].

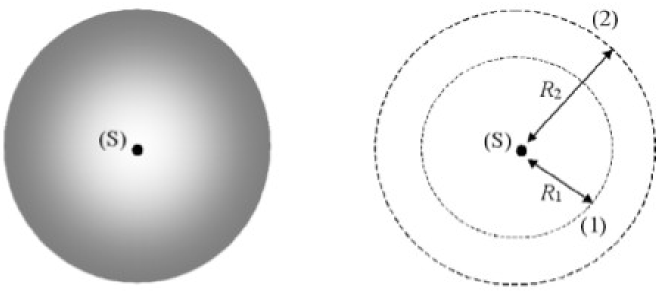


Figure 3.2: Spherical Spreading Loss [39]

The decrease in local acoustic intensity between points (1) and (2) is inversely proportional to the ratio of the surfaces Σ_1 and Σ_2 of the spheres [39] as expressed in equation 3.2 ,

$$\frac{I_2}{I_1}=\frac{\Sigma_1}{\Sigma_2}=\frac{4\pi R_1^{\ 2}}{4\pi R_2^{\ 2}}=\left(\frac{R_1}{R_2}\right)^2 \quad , \tag{3.2}$$

where R_1 and R_2 are the radial distances of point (1) and (2) from the source. The spreading transmission loss, considered from the reference unit distance (R_{1m} = 1m) can be expressed in equation 3.3 in dB as [39] ,

$$TL=20\log\left(\frac{R}{R_{1m}}\right) \quad . \tag{3.3}$$

3.1.4 Absorption

During propagation, wave energy may be converted into other forms and is absorbed by the medium. Certain properties of the medium act as imperfections for the propagation of a particular wave. Hence some energy is lost in overcoming those imperfections when the wave propagates. This is called absorption loss. The absorption loss for acoustic wave propagation underwater depends on frequency (f) and can be expressed as $e^{\alpha(f)d}$ where d is the propagation distance and $\alpha(f)$ is absorption coefficient at frequency f [6]. According to [39] in sea water, absorption occurs due to ,

a) pure water viscosity, the effect of which increases with squared frequency,

b) relaxation of magnesium sulphate ($MgSO4$) molecules below 100 kHz,

c) relaxation of boric acid ($B(OH)3$) molecules below 1 kHz.

The absorption coefficient at frequency f in kHz can be written as the sum of chemical relaxation processes and absorption from pure water as in equation 3.4 [39] [40] ,

$$\alpha(f)=\frac{A_1 P_1 f_1 f^2}{f_1^2+f^2}+\frac{A_2 P_2 f_2 f^2}{f_2^2+f^2}+A_3 P_3 f^2 \quad , \tag{3.4}$$

where the first term on the right side is the contribution from boric acid, the second term is from the contribution of magnesium sulphate, and the third term is from the contribution of pure water; A_1 , A_2 , and A_3 are constants; the pressure dependencies are given by parameters P_1 , P_2 and P_3 and the relaxation frequencies f_1 and f_2 are for the relaxation process in boric acid and magnesium sulphate, respectively. Figure 3.3 shows the relative contribution from the different sources of absorption as a function of frequency, and the variation in total absorption with frequency for four different oceans [40] .

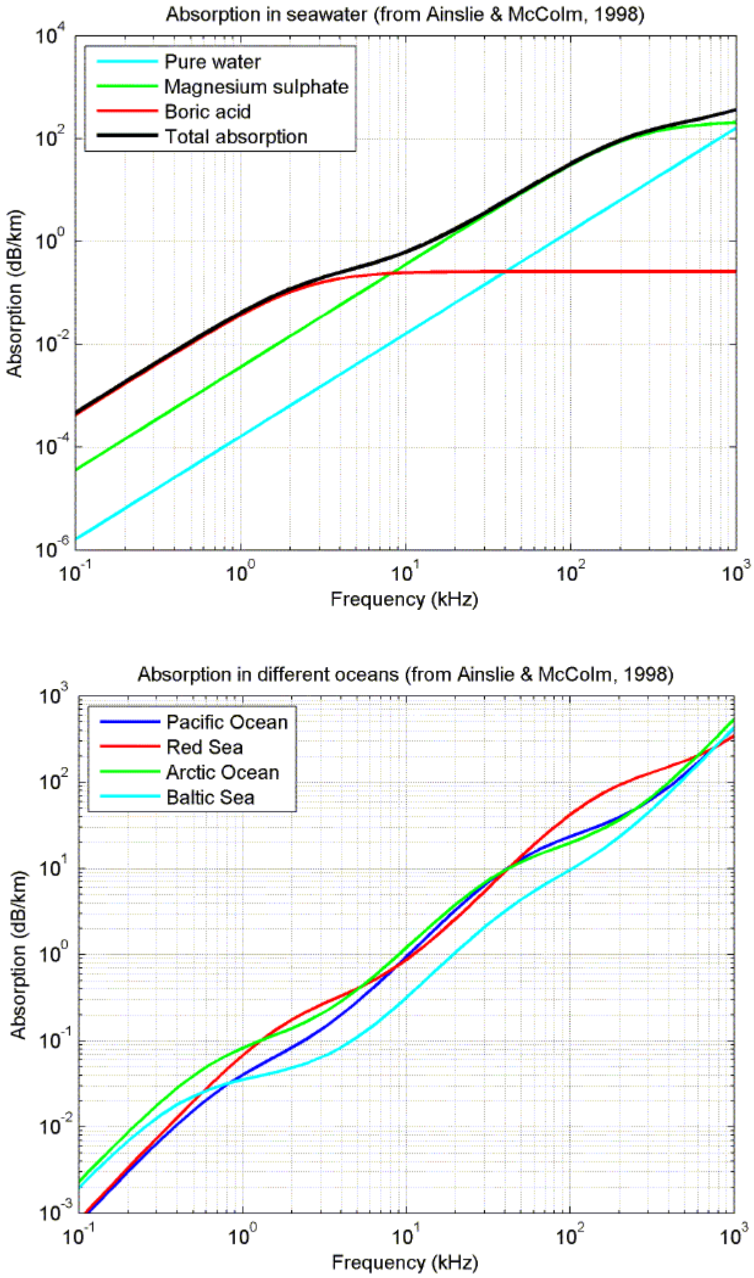


Figure 3.3: (Upper panel) Absorption in generic seawater, (Lower Panel) Absorption in different oceans [40]

3.1.5 Scattering Loss

Scattering loss is caused due to non-uniformities in the medium and adds to the multipath phenomenon (discussed in section 3.1.7). These non-uniformities (caused by particles and bubbles etc.) force some part of wave radiation to deviate from a straight trajectory. For example, deviation of a reflected wave from the angle predicted by the law of reflection is caused by scattering.

3.1.7 Multipath

An acoustic signal can reach a receiving antenna through multiple paths caused by reflections and refractions in the sea. Reflections can be from surface, objects and seabed, whereas refraction is due to varying density within seawater. The multipath phenomenon is shown in the Figure 3.5 below, with approximate dimensions: water depth 90m, horizontal range 1000m, source depth 15m, receiver depth 83m. Note that the first group of 4 arrivals, very clearly distinguishable, spreads over about 4ms; a second group arrives 20ms later, with still a very significant intensity, and a more blurred time structure, the individual echoes tending to spread and to merge into a reverberation trail [39] . Due to multipath an efficient modulation and demodulation of acoustic signal in underwater becomes challenging.

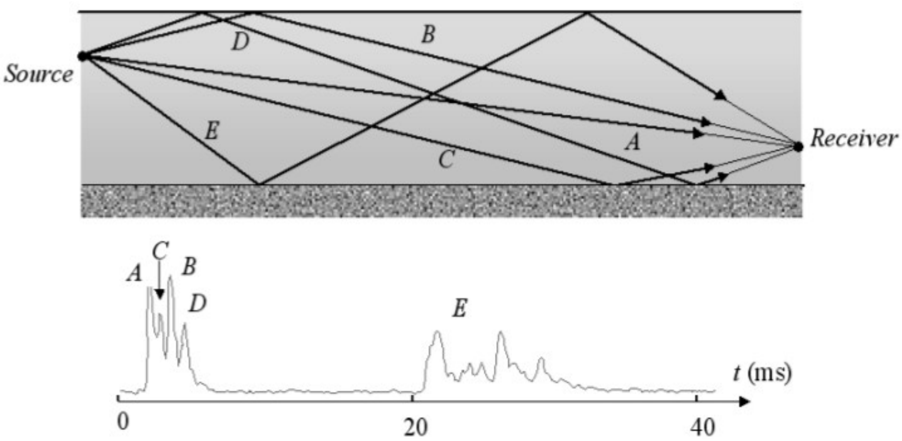


Figure 3.4: (Upper panel) Multipath trajectories in an isovelocity shallow-water configuration: (A) direct path; (B) reflection on the surface; (C) reflection on the bottom; (D) reflection on the surface and bottom; (E) reflection on the bottom and surface. (Lower panel) Multiple paths as visible in the envelope of real time- domain signal [39].

3.1.8 Doppler effect

The Doppler effect occurs when the frequency of a sound wave changes at a receiver due to relative motion of the transmitter or receiver. It has been commonly observed that the sound pitch becomes higher than the emitted frequency when the transmitter or receiver is approaching towards each other and lower than the emitted frequency when any of them moves away from the other. The magnitude of the Doppler effect *a* is proportional to the ratio of the relative transmitter-receiver velocity *v* to the speed of sound *c* [38] as given below in equation 3.5,

$$a = \frac{v}{c} \quad . \tag{3.5}$$

Doppler shift can be useful to track a target, and even help to identify it by measuring its speed. Figure 3.6 adopted from [39] , shows an acoustic transmitter passing by a fixed receiver, the transmitter has a linear speed *v* , and the minimum range between the transmitter and the receiver is represented by *H*.

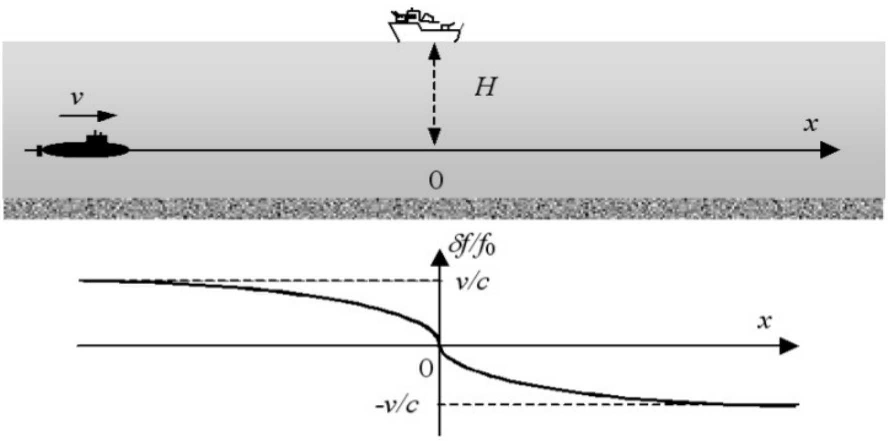


Figure 3.5: Effect of a Doppler shift between a mobile transmitter and a fixed receiver [39].

The radial speed component v_r [39] is expressed as,

$$v_r = \frac{-xv}{\sqrt{x^2 + H^2}} \quad , \tag{3.6}$$

which in terms of frequency shift, will be [39],

$$\frac{\delta f}{f_0} = \frac{-xv}{c\sqrt{x^2 + H^2}} \quad . \tag{3.7}$$

The last expression shows that the Doppler shift goes to zero and changes sign at the closest point of approach x = 0 as depicted in Figure 3.6. The relative frequency variations observed using sonar can be actually significant, i.e. $\delta f/f_0 \approx 0.7\%$ for a moderate relative speed of 10 knots (18.5 km/h), that is f_0 much larger than the typical variations encountered with radar (0.0002% for a plane travelling at 1000 km/h, the radar wave velocity being 3×10^8 m/s) [39] .

3.1.6 Noise

Noise in an acoustic channel comprises *ambient noise* and *site specific noise.* Ambient noise is present in the background of deep sea and is mainly caused by marine animals, turbulence, waves and distant shipping. Site specific noise exists only in certain places, for example, ice cracking in polar regions creates acoustic noise as do snapping shrimp in warmer waters [38].

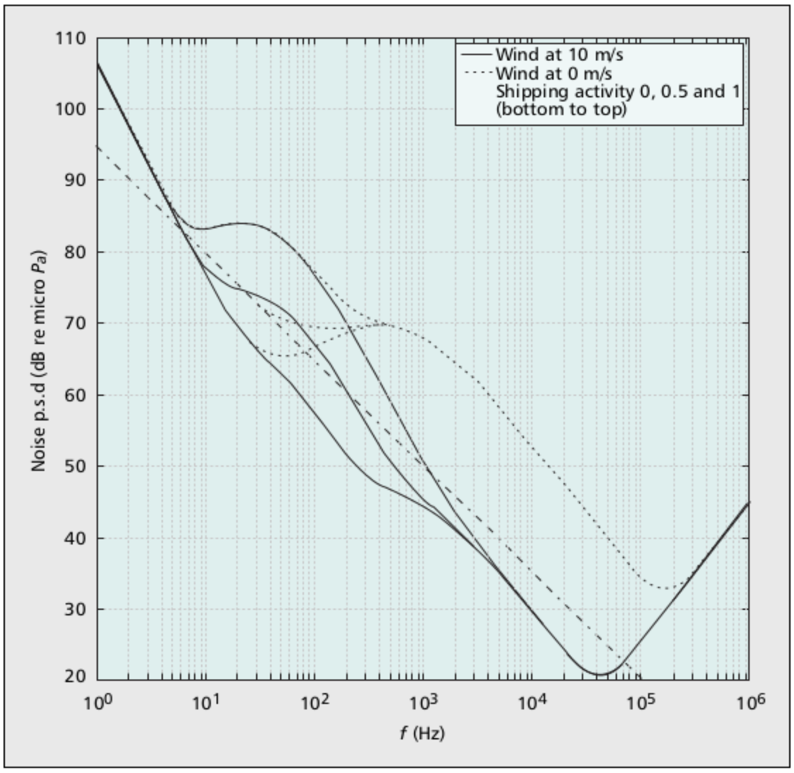


Figure 3.6: Power Spectral Density of Ambient Noise [38] [41].

Figure 3.4 shows the power spectral density of ambient noise for several values of wind speed (wind drives the surface waves) and several levels of distant shipping activity (which is modelled on a scale from 0 to 1). The power spectral density of ambient noise decays at a rate of approximately 18 dB/decade, as shown by the straight dashed line in Figure 3.4 [38] [41].

3.2 Electromagnetic Communication

The propagation of electromagnetic waves (EM) in the radio frequency range are often impractical in seawater due to their high absorption. They have only been proposed for short range links (1-10m) where their very high bandwidth (≥ MHz) can be exploited. Radio waves propagate through conductive salty water only at extra low frequencies (30− 300 Hz), which require large antennae and high transmission power.

EM waves possess faster velocity and higher operating frequency (which provides high bandwidth) as compared to acoustic waves, but seawater is a high-loss medium due to the cumulative increase of total dissolved solid (TDS) concentration. The TDS concentration increases the electric conductivity σ of seawater, which is about two orders higher than that of freshwater. In highly conductive medium, both the propagation velocity and the absorption loss of EM waves are functions of carrier frequency [6]. The propagation speed c of EM waves in seawater and absorption coefficient α can be approximated as in [42],

$$c \approx \sqrt{\frac{4\pi f}{\mu\sigma}} \quad , \tag{3.8}$$

$$\alpha \approx \sqrt{\pi f \mu\sigma} \quad , \tag{3.9}$$

where μ is the magnetic permeability and σ is the electric conductivity of sea water. From the two equations above, the propagation speed and absorption loss are approximately proportional to the square root of frequency. Hence using low frequency has been recommended in seawater due to high conductivity. A plot of the velocity and the absorption coefficient versus frequency for EM waves in seawater is shown in Figure 3.7 [6].

The ratio of electric conductivity σ and dielectric permittivity ϵ, called transition frequency *(σ/ϵ)* of a medium defines the behaviour of an EM wave in that medium. If the frequency of an EM wave is lower than transition frequency of the medium it behaves mostly like a diffusion field; if the frequency is higher than the transition frequency then it behaves like a propagating wave [6]. For seawater, the conductivity σ is about 4 S/m, and the dielectric permittivity is $81\times10^{-9}/(36\pi)$. These values yield a transition frequency of about $4 \times 36\pi \times 10^{9}/(2 \times 81\pi) = 888$ MHz [6]. This means that if an EM carrier wave frequency is 10 MHz then it will no longer behave like a wave rather a diffused field however on the other end of spectrum, if a carrier with a frequency of 1 GHz is used, the EM field will mostly behave like a wave. But, due to the high absorption of seawater (see Equation 3.9 above), the EM wave can hardly propagate [1] .

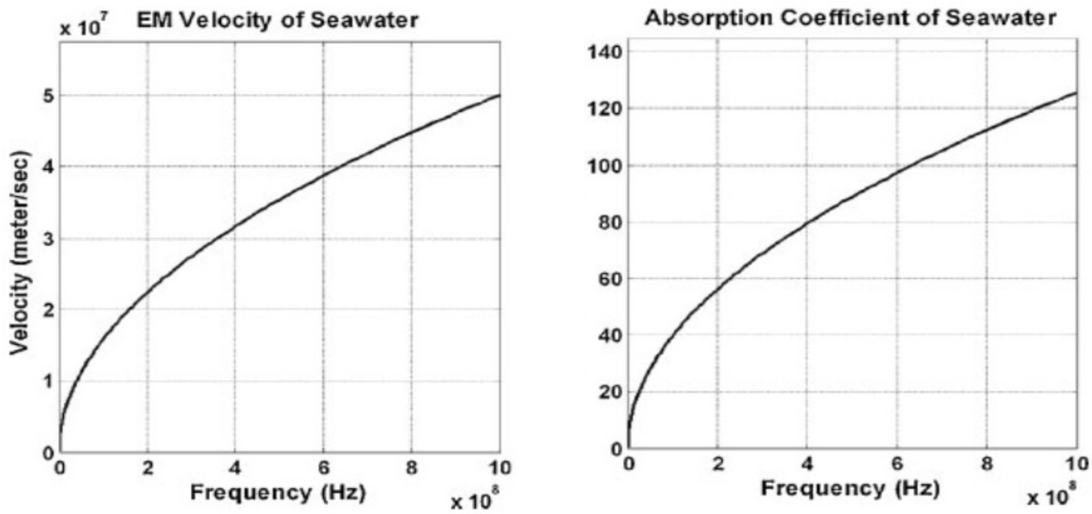


Figure 3.7: Velocity and absorption coefficient verses frequency for EM waves in sea water [6] .

Contrary to this it is shown in [43] that although EM wave propagation has high attenuation in the near field, there exists a far field region that would allow transmission over longer distance than previously thought. Propagation through seawater at a frequency of MHz over a distance of 85m appears to be possible [44]. However, if EM could be working underwater, even in a short distance, its much faster propagating speed is definitely a great advantage for faster and efficient communication among nodes [6]. Newer subsea radio frequency (RF) communications systems that use digital technology and signal compression are capable of data rates of 100bits/s up to 100m and up to 100kbit/s over 10m. These data rates potentially make the technology useful for subsea wireless sensor networks for production monitoring applications. However, these systems have high power requirements on both transmit and receive sides, making them less practical in a deployable wireless solution.

Recently preliminary reports have surfaced about higher-frequency subsea RF systems that use a less well-known feature of through-water transmission (Debye relaxation). Although the near-field energy of such a system is absorbed rapidly by the water, a weaker far-field component with, for example a 5 MHz frequency, may be detected over tens, and potentially hundreds of meters.

Subsea RF communications are immune to acoustic noise but can be impacted by salinity variations in the sea water and by subsea RF interference from electrical equipment. In developing an RF wireless network for hydrocarbon production systems, it is important to account for complex metal structures and to ensure that the wireless network design minimizes potential problems associated with interference and reflection [45]. In [22], the authors have presented the lessons learned from two years of operation of radio communication based wireless sensor networks deployed at seven coral reefs along the Great Barrier Reef in north-eastern Australia and have clearly highlighted the impractical approach of using radio communication based wireless sensor networks for underwater applications. They finally proposed a practical, real-time, acoustic communication based UWSN for monitoring of coral reefs. The system design considers the harsh acoustic communication environment and performs reliable marine life monitoring.

3.3 Optical Communication

Free-space optical (FSO) waves used as wireless communication carriers are generally limited to very short distances because of the severe water absorption at the optical frequency band and strong backscatter from suspending particles [6]. Even the clearest water has 1000 times the attenuation of clear air, and turbid water has more than 100 times the attenuation of the densest fog [6]. Nevertheless, underwater FSO, especially in the blue-green wavelengths, offers a practical choice for high-bandwidth communication (10-150 Mbps) over moderate ranges (10-100 meters). This communication range is much needed in harbour inspection, oil- rig maintenance, and linking submarines to land, just to name a few of the demands on this front [6].

Recent advances in light emitting diode (LED) and laser diode technology have produced efficient, compact, long life time optical sources emitting in this range. Similarly, there are a number of efficient optical detectors for these operating wavelengths. In addition to the band widths achievable from LED or laser diode sources, the choice of optical source for a particular application depends on the angular spread, or beam divergence. For high bandwidth point-to-point communications, laser diodes generally are preferable. For lower bandwidth applications involving multiple nodes, the greater angular spread of LEDs may make them preferable [45].

The scattering process of optical waves and the wavelength dependence of underwater optical channels can be evaluated by the Mie scattering theory [46] . According to the Mie theory [46] , when the light wavelength is similar to the particle diameter, light interacts with the particle over a cross-sectional area larger than the geometric cross section of the particle . The attenuation due to optical scattering can be expressed as in [46] by,

$$\frac{dI}{dx} = -\zeta I \quad , \tag{3.10}$$

where I is the intensity of light and ζ is the turbidity in the sea. Turbidity is a measure of the amount of cloudiness or haziness in seawater caused by suspending particles, it provides an indication of the clarity of the seawater and is measured using the nephelometric turbidity units (NTU) [6]. Seawater possesses turbidity ranging from tens to several thousands of NTU [47]. After solving the above differential equation about multi-scattering light intensity we get a solution very similar to EM absorption [6],

$$I_d = I_o e^{-\zeta d} \quad , \tag{3.11}$$

where I_d is the light intensity at distance d through the medium with multiple scatterers and $I_0$is the incident light intensity, the role of the turbidity ζ is exactly the same as the absorption coefficient α in wave absorption loss [6]. Hence sea turbidity ζ is an important factor in deciding the propagation of an optical wave in sea. Table 3.1 [6] below summarizes the comparison of acoustic, electromagnetic and optical wave propagation in the underwater environment.

Table 3.1: Comparison of acoustic, electromagnetic and optical wave propagation in underwater [6]

	Acoustic	Electromagnetic	Optical
Nominal Speed (m/s)	~ 1500	~ 3.0×10^8	~ 3.0×10^8
Power Loss	> 0.1 dB/m/Hz	~ 28 dB/1km/100Mz	∝ turbidity
Bandwidth	~ kHz	~ MHz	~ 10- 150 MHz
Frequency band	~ kHz	~ MHz	~ 10^{14}-10^{15} Hz
Antenna Size	~ 0.1 m	~ 0.5 m	~ 0.1 m
Effective range	~ km	~ 10 m	~ 10- 100 m

3.4 UWSN Protocol Stack

Being a resource constrained system and having acoustic waves as the main physical layer technology, UWSN faces challenges at different layers of the communication protocol stack. Typically we consider five layers of OSI (Open System Interconnection) standard in UWSN i.e. Physical layer, Data Link/ Medium Access Control (MAC) layer, Routing layer, Transport layer and Application layer. Most challenges are faced due to the physical mode of wireless propagation and underwater environment effect and have impact on the physical, MAC and routing layer mainly.

3.4.1 Physical Layer

This layer is responsible for converting the data to wireless signals through encoding, modulation, error correction and channel equalization. The basic challenge is to design a spectrally efficient modulation scheme and utilize the limited bandwidth efficiently. Some highly effective techniques are developed and being worked on as discussed in [2] . Non-coherent modulation schemes are suitable for moderate data rates, high power efficiency and robust performance. With advancements in phase tracking algorithms to overcome the rapid phase variations in the underwater acoustic medium, phase coherent systems have become practical means for achieving higher data rates over different underwater channels, including severely time-spread horizontal shallow water channel [48] [49] . However, primarily real-time systems were employed for applications in vertical and very short range channels with stable phase and minimal multipath effects [1]. Further the channel equalization and error correction techniques are used to enhance the modulation.

3.4.2 Medium Access Control (MAC) Layer

It takes care of all the 1-hop away communication issues involving medium access control which is the wireless medium sharing and hence called the MAC layer [7]. The main task is to avoid collisions. In wireless sensor network due to several nodes participating in the communication, frequency spectrum is inherently shared and interference needs to be properly managed. While designing a MAC protocol for UWSN, the specific acoustic channel characteristics and hardware limitations need to be considered. These include propagation delay, frequency-dependent attenuation, relatively long reach of acoustic signal and the limited bandwidth of the acoustic channel. MAC layer is also responsible for error detection, data framing and other 1-hop away communication tasks.

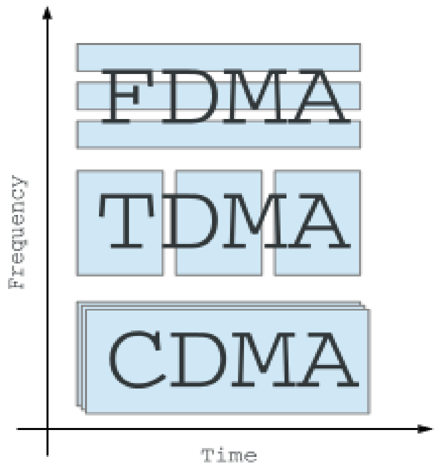


Figure 3.8 : Basic illustration of schedule-based MAC Protocols [50]

Traditionally MAC protocols try to minimize collisions by slotting the time (Time Division Multiple Access (TDMA)) or by sensing the channel prior to transmission (Carrier Sense Multiple Access (CSMA)). However due to long propagation delay of underwater transmission these networks also suffer from space uncertainty, hence it is necessary to take into account the locations of the receivers and their possible interferers [50]. This problem is commonly known in the literature as space-time or spatio-temporal uncertainty [51]. Another problem that comes alive due to long propagation delay is spatial unfairness [51] . As the packet reception time depends on the distance to the transmitter, the channel becomes free first at the transmitter and later on at the receiver. Hence the nodes closer to the transmitter are able to gain access to the channel before nodes closer to the receiver [50] .

Despite these problems MAC protocols can still be subdivided classically into two main categories i) schedule-based and ii) random-access based. In schedule-based schemes as shown in Figure 3.8, such as TDMA (time-division multiple access) and FDMA (frequency-division multiple access), each node is assigned a different time slot or frequency band, hence the nodes do not compete with each other to gain access to the communication medium. In random-access based schemes such as ALOHA and CSMA, there is no pre-allocation of resources, thus allowing the nodes to compete with each other to gain the medium access on demand. Moreover, CDMA (code-division multiple access) based MAC protocols can be used in both scheduled and random-access based environments and possibly improve the system performance by allowing simultaneous code-division transmissions from multiple stations [2].

Table 3.2 summarizes the pros and cons of each category of MAC protocol for underwater acoustic (UW-A) communication. FDMA divides the available bandwidth into different bands, allowing different nodes to transmit and receive at the same time. However, the narrow bandwidth of UW-A channel and limited bandwidth of underwater systems which are prone to multipath and fading, makes this scheme rarely used in underwater networking. Similarly, TDMA schemes have been proposed for underwater environment but they show limited channel utilization in large-scale networks. As shown by the Seaweb project [52], where three clusters were deployed, FDMA was used for inter-cluster communication, while TDMA was used within each cluster. But soon it was realized that the bandwidth was being used inefficiently and the protocol was vulnerable to fading and multipath. Therefore, current underwater MAC solutions are for the most part based on random access schemes such as ALOHA, CSMA or CDMA [2].

Table 3.2: Classification of MAC protocols in underwater communications [2]

	Pros	Cons
FDMA-based	Multiple users access simultaneously	Narrow bandwidth in UW-A channels and vulnerability of limited band systems
TDMA-based	Avoiding collisions	Limited channel utilization efficiency in large-scale networks
ALOHA-based	Easy to implement	Pure ALOHA has limited channel utilization
CSMA-based	Prevents collisions with ongoing transmission	Channel may be sensed idle while a transmission is ongoing
CDMA-based	Robust to frequency-selective fading caused by underwater multipaths	Near-far problem reduces the performance

ALOHA based MAC protocols have been refined over the years for better throughput. Pure ALOHA has low efficiency [53] compared to its improved version Slotted ALOHA, in which discrete time slots have been introduced and data is sent only in the beginning of a time slot hence reducing the collisions and increasing the throughput. However, two more worth mentioning ALOHA based protocols ALOHA-AN (ALOHA with advance notification) which is better version of ALOHA-CA (ALOHA with collision avoidance) are presented in [54] . In ALOHA-CA sender-receiver information and propagation delay of the packet is used by each node to decide when to transmit its packet to avoid collision. Each packet has a header and a data segment and each node monitors its neighbouring nodes to update its database. Before sending a packet, the data base is checked to avoid collisions. In ALOHA-AN a node sends a small advance notification (NTF) packet, then it waits for a lag time before sending the actual data packet. Due to lag time, a receiving node extracts information from different NTF packets and makes better decisions in order to avoid collisions. Careful selection of lag time in ALOHA-AN offers better efficiency and throughput than ALOHA-CA.

Conventional CSMA based MAC protocols are not suitable for UWSN. Due to the high propagation delay, the channel may be sensed idle while a transmission is going on. However slotted FAMA (floor acquisition multiple access) [55], a CSMA based MAC protocol, combines carrier sensing and synchronization of source and receiver to reduce the probability of collisions. However its efficiency is affected by longer delays due to time guards. A comprehensive comparison and performance of CSMA-based MAC protocols is discussed in [56]. The performance was evaluated during extensive practical tests. The results show that larger packet sizes can lead to significantly better system performance in terms of throughput efficiency, at the cost of increased packet latency, especially for low traffic loads [2]. The authors in [56] also emphasize how acoustic modem limitations affect the at-sea performance and any improvement in the hardware can significantly improve the network performance.

CDMA based MAC protocols are robust to frequency-selective fading caused by multipath in underwater transmission. A distributed MAC protocol named UW-MAC [57] for UWSN is worth mentioning. It is shown by extensive simulations that it achieves high network throughput, low channel access delay, and low energy consumption, all three objectives simultaneously in deep water, which is usually not prone to multipath. However in shallow water communications, which can adversely be affected by multipath, it dynamically finds the optimal trade-off among these objectives according to the application requirements [2] .

Multiple-input-multiple-output (MIMO) techniques which involve the use of multiple antennas at the transmitter as well as receiver end offer improved performance compared to single-input-single-output (SISO) systems. They may exploit the rich scattering and multipath fading to provide higher spectral efficiencies without increasing power and bandwidth [2] . MIMO systems benefit from the transmission rate increasing with the multiplexing gain and the BER decreasing with increasing diversity gain. A new MIMO based MAC protocol called UMIMO-MAC is proposed in [58] . It is designed to adaptively trade-off between multiplexing and diversity gain depending on the application requirements and channel conditions.

3.4.3 Routing Layer

The basic aim of this layer is to achieve the optimal hop-to-hop path and update that ideal route. However due to water current and harsh environment this becomes extremely challenging. This layer has been investigated intensively and several routing protocols have been developed. However, due to the lack of a standard simulation and experimental platform, it is difficult to compare them analytically. There are two main categories of routing protocols; proactive and reactive. *Proactive routing* protocols try to maintain the up-to-date routing information at all nodes at all times, hence minimizing the delay in data transmission. However in *reactive routing* protocols, the route is discovered only when an event occurs and a route from source to destination is required, hence minimizing the energy consumption but delaying the data transmission. For UWSN, proactive routing has large signalling as well as energy overhead since the network topology changes continuously due to water current and node mobility. Scalability and excessive use of bandwidth is a major issue making them unsuitable for dynamic UWSN. On the other hand, the reactive routing in the underwater environment has large delays which are amplified by slow propagation of acoustic waves, hence communication is adversely affected. There is a third category, geographical routing, which is localization based. However, since GPS doesn't work underwater, localization is an issue in itself for UWSN, to be dealt with efficiently. Improving the routing performance of an UWSN is very much needed in order to ensure proper sensing and transmission coverage as well as connectivity. Providing a network with several back-up paths for this purpose can bring significant improvement as the disconnectivity due to changing network configuration can be over come by back-up routes.

The routing protocols for UWSN are discussed in detail in Chapter 4. Based on transmission technique, the flooding based routing approaches for UWSN (such as HH-VBF [59], FBR [60], DBR [61], H^2-DAB [62], SBR-DLP [63] etc.) are simple at computation and have acceptable end-to-end delay, low processing overhead and can be utilized for delay sensitive applications [64]. However, the energy consumption in duplicate transmission, congestion and channel sharing remains a problem [35]. This has been tried to overcome by some specific individual approaches by each protocol to some extent. However the existence of VOID regions (a region which is created due to random deployment, node movement or failure and is not covered by any other node) and changing network configuration limits this approach. Some of the flooding based approaches involve localization, another energy overhead, as the network configuration is constantly changing with water current.

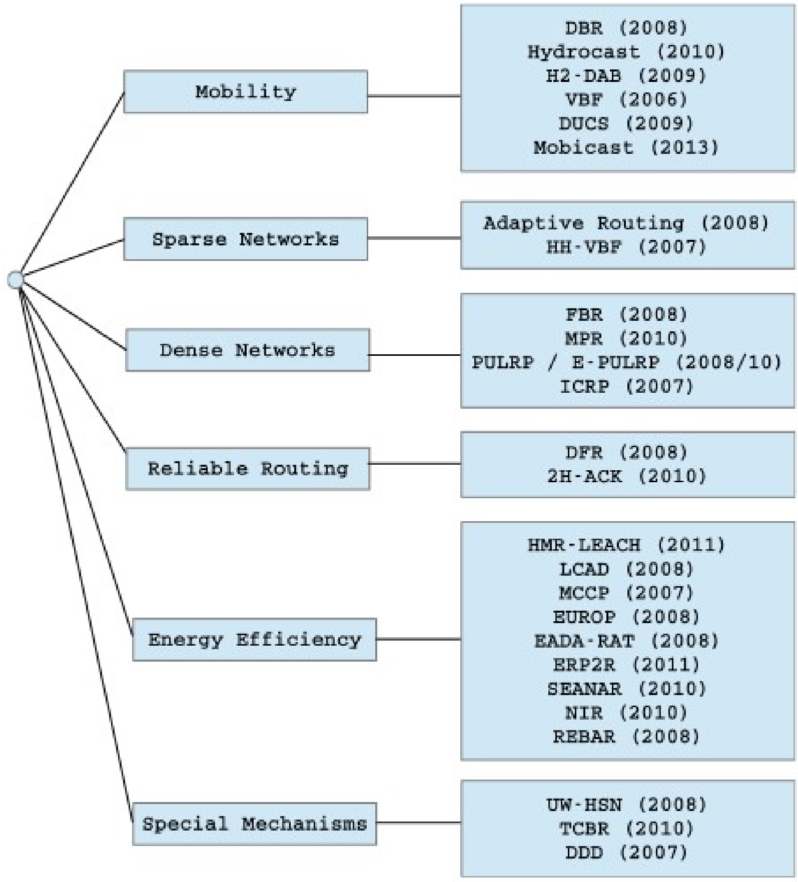


Figure 3.9: Classification of Routing Protocols based on application requirement and performance [50]

The protocols utilizing multi-path transmission (such as MVSA [65], RRA [66] and REBAR [67]etc. compared to flooding based approaches offer acceptable reliability, energy efficiency, low end-to-end delays, less congestion and interference but more computation involved in maintaining more paths [68]. Therefore, efficient solutions are needed to avoid repetitive transmission. In addition the changing network configuration has been completely ignored and this will cause loss of connectivity in a sparse network [35]. The clustering based protocols (such as DUCS [69], Pressure Routing [70] and DCB [71] etc.) are complex in terms of processing overheads due to node mobility and re-clustering. An efficient routing mechanism for re-clustering can reduce these overheads and this area needs to be worked on further in order to avoid loss of connectivity due to changing network configuration. A dynamically reconfigurable routing protocol is in high demand. Based on performance and application requirement some basic classification of UWSN routing protocols is shown in Figure 3.9 [50], however, many protocols could belong to several classes.

3.4.4 Transport Layer

This layer is responsible for reliable end-to-end (E2E) communication by retransmission and time out checks. Also, it is responsible for reducing the network congestion. However, this layer is omitted at relay nodes due to multi hop routing strategy. In UWSN due to changing network configuration we try to ensure end-to-end transmission at each hop.

3.4.5 Application Layer

All activities concerning the display of data at the receiver end are taken care of in this layer. However this layer is omitted at the sending node and relay node. The limitations of UWSN and unpredictable nature of underwater channel has an impact on the design of application layer for the network. Task specific requirements of the network supports cross-layer design in application layer.

3.5 UWSN Simulators

Underwater wireless sensor networking experiments in the field using the real hardware are very expensive. To facilitate researchers a number of simulators have been developed. However most of them lack the complexity of physical layer hence the simulation results can differ from the real in-field results significantly. An Intermediate solution is testbeds (discussed in section 3.6) , some of which have recently been made available to other researchers. In addition there are some frameworks which support the user all the way from the simulation to hardware testing. Here is a summary of some of the simulators.

3.5.1 AquaSim

It is an ns-2 based UWSN simulator that claims to effectively simulate acoustic channel. It supports 3D and mobile networks [72] . Aqua-Sim can easily be integrated to ns-2 and it is independent of any changes to ns-2 or vice versa. The comparison of simulation results and real testbed results is shown below in figure 3.10 and 3.11 [72] .

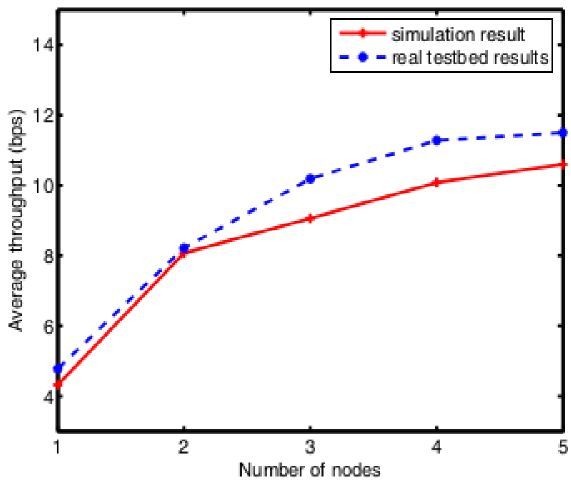


Figure 3.10: Throughput with fixed input traffic per node [72]

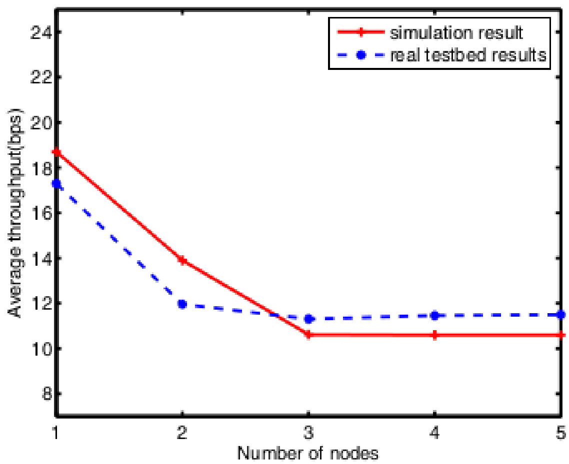


Figure 3.11: Throughput with fixed total input traffic [72]

Aqua-Sim follows the object-oriented design style of ns-2, and all Compost Bin network entities are implemented as classes in C++. Figure 3.12 shows the class diagram of Aqua-Sim, in which the “UnderwaterNode” object is the abstraction of the underwater sensor node, the “UnderwaterChannel” object represents the underwater acoustic channel which provides public interface to upper layers and thus the object in the upper layer, such as a routing layer object, can easily get to know the channel properties [72] .

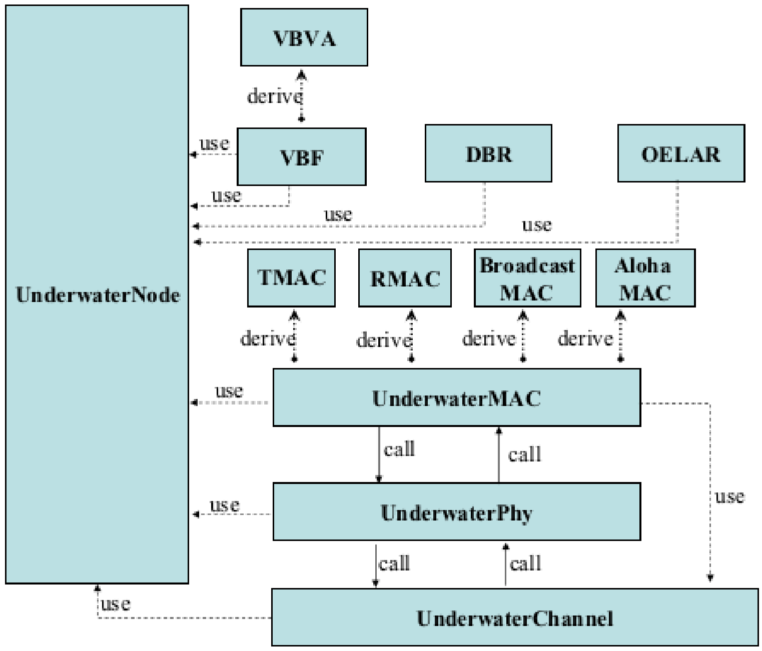


Figure 3.12: Class Diagram of Aqua-Sim [72]

3.5.2 WOSS

WOSS (World Ocean Simulaton System) [73] is an open source, multi-threaded C++ framework that integrates the Bellhop ray-tracing program [74] [75] [76] to provide a realistic propagation of sound in water instead of using empirical equations. The user only has to specify the location in the world and the time where the simulation should take place, this is done by setting the simulated date and the desired latitude and longitude of every node involved [73]. Then the simulator automatically handles the rest and provides a channel realization. WOSS is a powerful tool, that makes it possible to employ realistic underwater channel patterns within network simulators [77]. Initially it was developed to complement the popular network simulator ns-2 and its extension NS-MIRACLE (Network Simulator- Multi-InteRfAce Cross-Layer Extension), however it is a stand-alone package that can be interfaced to any simulation tool.

It provides an acoustic propagation model that is very accurate in the tens kHz frequency range, providing results that may not substitute experimental result but may represent reasonably realistic scenarios [78]. In addition, WOSS provides a set of routines to facilitate many standard operations, like mobility management, the conversion among different co-ordinate systems (e.g., Cartesian to spherical), and the maintenance of the data structures required for the simulation of acoustic propagation [77]. However, due to high complexity the simulation is limited to small network and low number of nodes.

3.5.3 DESERT

DESERT (short for DEsign, Simulate, Emulate and Realize Test-beds for Underwater network protocols) [79] is an open source complete set of C++ libraries that extend the NS-MIRACLE simulator to support the design and implementation of underwater network protocols from simulation to emulation to at-sea testing . It was initially developed in 2012 however version 2 of DESERT has been released in April 2014 which is more robust and user-friendly. DESERT Underwater extends the NS-MIRACLE [79] simulation software library, developed at the University of Padova, in order to provide several protocol stacks for underwater networks, as well as the support routines required for the development of new protocols [79] . One of the main features of DESERT underwater simulator is that it allows using the same code both for simulations and the real hardware (at-sea) experiments and this has been extended to a broad range of devices in the latest version. It can also benefit from the use of WOSS in order to get the simulation results over a realistic channel model. The functionalities offered by DESERT have been extended with a remote control framework called RECORDS, that provides a set of primitives to remotely control the hardware modems, and thus the network operations [77] .

3.5.4 SUNSET

Similar to DESERT, SUNSET [80] (Sapienza University Networking framework for underwater Simulation Emulation and real-life Testing) is another promising open source framework based on ns2 and NS-MIRACLE. SUNSET was initially released in 2012 and version 2 was released in late 2013 with improvements specially in the channel and interference models as well as hardware compatibility. SUNSET also benefits its users by allowing them to use the same code for simulation, emulation and at- sea testing and has been validated through several sea trials. Similar to RECORDS in DESERT, SUNSET has a feature called *back-seat driver* , to remotely control and operate the entire underwater network via acoustic links [80] [81] .

3.5.5 UWSim

UWSim is an open source simulation tool for 3D visualization and simulation of UWSN which is based on network component approach rather than OSI layer based approach. The 3D scene is highly configurable by the user, that first sets the basic geometry using third-party modelling software (like Blender or 3D Studio Max etc), and then can include multiple simulated underwater vehicles and manipulators [82] . UWSim can be used mainly as validation of external control programs that can be easily attached to the virtual world and simulation of sensors such as virtual cameras and range sensors [82] . The interface with external software is implemented through ROS (Robot Operating System) interfaces, thus allowing it to be used in a distributed manner, also vehicle dynamics can be simulated and interfacing with Matlab is possible [82] .

3.5.6 AUVNetSim

AUVNetSim [83] is a simulation library for testing acoustic networking algorithms, written in Python and makes extensive use of the SimPy discrete event simulation package. AUVNetSim is interesting for both end users and developers; as a user willing to run several simulations using the resources that are already available, can easily modify several system parameters without having to explicitly deal with Python code, and a developer who wants to include a new MAC protocol can simply do so by taking the advantage of the existing structure [83] . However the physical layer in AUVNetSim is too simple based on Thorp (a simple acoustic propagation model which shows signal attenuation only with respect to distance without taking into account the depth or radiation pattern of the source) approximations, so different environment conditions cannot be considered in the propagation model, leading to simulation results that may be far from the ones obtained in real network deployment [84] .

3.6 Experimental TestBeds

To avoid the expensive nature of underwater wireless sensor network experimentation in real sea, and the simplicity of physical layer designed in some simulators, there is an intermediate solution, a step between simulation and at sea experimentation which is the testbed. Some of the testbeds are discussed below.

Seaweb [52] one of the first experimental platforms run by Spawar Systems Centre (SSC) San Diego and Naval Postgraduate School (NPS), funded by Office of Naval Research (ONR), US, with Teledyne Benthos as the main contractor [2]. Seaweb consists of autonomous backbone nodes, stationary sensor nodes, telesonar repeaters and peripheral nodes which include unmanned undersea vehicles (UUVs) and low-frequency sonar projectors. In past many networking protocols have been field tested to validate their performance using Seaweb.

The Centre for Maritime Research and Experimentation (CMRE) [85] is a scientific facility of NATO which also maintains and runs an underwater networking testbed with heterogeneous modems [2] [86] .

Ocean-Tune [87] is an open testbed suite which resulted from the collaborative work of University of Connecticut, University of Washington, University of California, LA and Texas A&M University in order to support the research community in networking, communication, engineering and marine technology. Ocean-Tune is comprised of four testbeds remotely accessible to public at four different sites which provide flexibility in the choice of surface nodes, bottom nodes and mobile nodes (gliders and drifters). These nodes are equipped with OFDM (Orthogonal frequency-division multiplexing) acoustic modems which facilitate high data rates and strong networking capability.

DESERT and *SUNSET* (as discussed in section 3.5) are also worth mentioning here as testbeds as they both support the user through their journey from design, simulation to experimental testing .

WHOI (Woods Hole Oceanographic Institution) and University of Connecticut have developed an underwater acoustic network (UAN) testbed [88], which can be made available for collaborative experiments with the UAN research community. For most experimental configurations the nodes can be remotely controlled via internet [88].

University at Buffalo has made an underwater acoustic testbed, *UW-BUFFALO* [2] as a joint venture with Teledyne Benthos. It is based on the Teledyne Benthos Telesonar SM-75 modem which in its different configurations is a key component in multiple U.S Navy programs and in many wireless tsunami warning systems worldwide

[2]. This modem on its own cannot be configured by the user, however the developers of *UW-BUFFALO* have modified it and have added components to allow the research community to perform advance networking and communication experiments. With the SM-75 modem they have interfaced a programmable Gumstix network processor, a reconfigurable software defined protocol stack including medium access control, IP network layer with reconfigurable ad hoc routing. In addition the network self configuration primitives (e.g., neighbour discovery, DHCP), are being implemented on the Gumstix board to enable the definition of complex networking experiments with reconfigurable, cross layer designed protocol stacks [2]. Furthermore the modified platform allows playing and recording custom defined acoustic waveforms to allow reconfigurable physical layer experimentation with arbitrary transmission schemes [2].

3.7 Summary

In this chapter the underwater wireless communication techniques (acoustic, electromagnetic and optical) are discussed. Acoustic communication characteristics like propagation speed, path loss, geometric spreading, absorption loss, scattering loss, multipath, Doppler effect and noise are explained from an underwater wireless sensor networking point of view. Then the features of electromagnetic and optical transmission in underwater environment are presented. Discussing all the wireless transmission techniques, it is highlighted that acoustic propagation is the preferred mode of wireless communication in underwater environment. After this the UWSN protocol stack is discussed with five layers communication i.e. Physical layer, MAC layer, routing layer, transport layer and application layer. Finally, an overview of the simulators and testbeds available for experimenting an UWSN are presented.

CHAPTER 4

Routing Protocols for UWSN

4.1 Introduction

Similar to a WSN, the performance of an UWSN is highly dependent on the efficiency of the networking protocols and hardware involved. One of the important research areas in UWSN technology is the design, development and improvement of routing protocols. A routing protocol ensures reliable data transmission from source node to the sink node. Routing protocol design in UWSN is far more challenging than in terrestrial WSN because of the unique nature of acoustic waves transmission in the underwater environment (low propagation speed, limited bandwidth capacity, high error probability, etc.). In addition, the 3D underwater environment and the continuous node movement due to water current makes it even more difficult.

There are a number of routing protocols developed for UWSNs as discussed in [89] up till year 2011 and in [90] up till year 2016, most of them seem promising but offer a lot of room for improvement. One major common flaw is that the performance evaluation of these protocols is done either in 2D or in 3D using random node deployment strategy. The underwater environment being 3D, a 2D node deployment is not applicable and 2D results cannot be interpreted for 3D. Underwater sensor nodes are bigger in size than terrestrial WSN nodes and are expensive. Random node deployment will be a waste of energy as well as network resources as it creates dense as well as sparse regions in the network and chances of VOID (a gap, an empty space or a region which is not covered by any node) creation is high. Due to water current and node movement this is an extremely difficult issue to deal with. Hence random node deployment in 3D cannot ensure reliable transmission of data with optimum use of resources. A comprehensive 3D node deployment strategy to ensure sufficient back-up paths to combat the effect of node movement due to water current, for reliable transmission of data is needed. This is discussed in detail in Chapter 5. Here in this chapter we present briefly some of the promising routing protocols, whose performance can be improved significantly by introducing a 3D node deployment strategy which is simple to implement and yet effective. Most of these routing protocols do not offer sufficient reliability for the transmission of data in sparse network or address the changing network situation due to node movement. Hence our strategy overcomes these issues. The routing protocols for UWSN are divided into four categories for understanding; however some of them can belong to two or more categories. These categories are: flooding based, multipath based, clustering based and miscellaneous.

4.2 Flooding Based Routing Protocols

In flooding based protocols a packet is transmitted to all the nodes in the transmission range. A flooding approach is simple and requires little knowledge about the network. The main issue relates to the duplicate packet transmission, congestion, channel interference, and the energy consumed in dealing with these issues. Some of the flooding based approaches are:

(i) Vector Based Forwarding protocol

(ii) Focused Beam Routing

(iii) Depth Based Routing

(iv) Hop by Hop Dynamic Addressing Based

(v) Directional Flooding based Routing

(vi) Sector Based Routing with Destination Location Prediction (SBR-DLP)

4.2.1 Vector Based Forwarding protocol (VBF)

VBF [91] is a localization based routing protocol. It establishes a virtual pipe around the vector from source node to sink node and floods the data packet in that pipe as shown in Figure 4.1. In a basic VBF all nodes inside the virtual routing pipe are qualified to forward the data packet. However in dense networks with so many forwarding nodes within the routing pipe, this will not only increase congestion but energy overhead is also increased. So the authors in [91] introduce a *desirableness factor,* which is the suitableness of a node to forward the data packet and *self adaptation algorithm* by which, the most desirable node is selected to forward the data packet. The nodes choose the most desirable (best) forwarding nodes among them by the *self adaptation algorithm* to forward the packet. A node which is at the edge of transmission range of the forwarding node and nearest to the routing vector, qualifies as being the most desirable node to forward the packet. Depending on the *desirableness factor* of each node, it delays forwarding the data packet in a manner that the most desirable node forwards the data packet first. Other nodes delay forwarding this packet hence reducing the duplicate transmission and if they receive this packet again during this delay than they re-evaluate their desirableness factor so as to drop the data packet if it has already been forwarded.

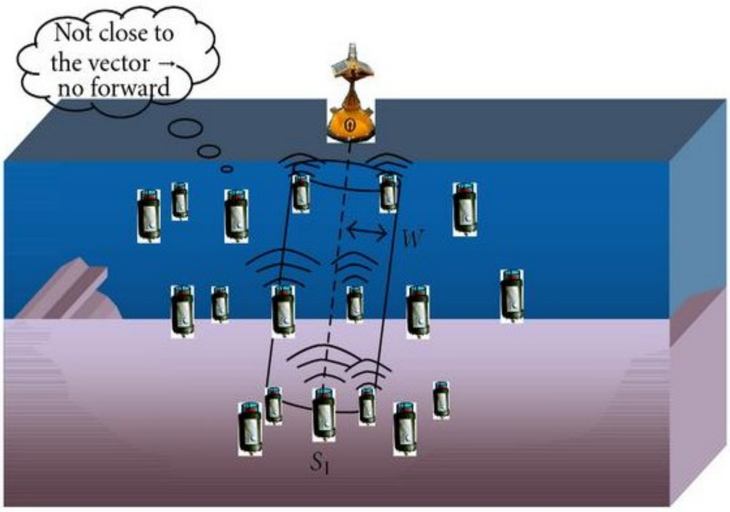


Figure 4.1: VBF routing protocol [91]

VBF performs better in dense networks. In sparse networks there might be a situation where there will be forwarding routes available outside the routing pipe and none within the routing pipe. Hence hop-by-hop VBF (HH-VBF) [59] is introduced which is an enhanced version of VBF. In HH-VBF a virtual routing pipe is formed at each hop as shown in Figure 4.2. At each hop the forwarding node forms a pipe around the vector from itself to next forwarding node and so on to forward the packet till it reaches the sink. Hence a virtual forwarding vector at each hop is established. In this way HH-VBF performs better than VBF in finding desirable forwarding nodes in a sparse network.

In [91] the performance of VBF is evaluated by randomly distributed sensor nodes in a 3D field and it is shown that self-adaptation algorithm based VBF performs better than basic VBF in terms of packet delivery and energy consumption (which is as expected). However, comparing the simulation results of VBF and HH-VBF it is revealed in [91] that HH-VBF has much better packet delivery than VBF; however, more energy is consumed as compared to VBF in order to maintain virtual routing pipes at each hop. Introducing node mobility, it is shown that packet delivery ratio reduces, however, energy consumption remains the same for both VBF and HH-VBF.

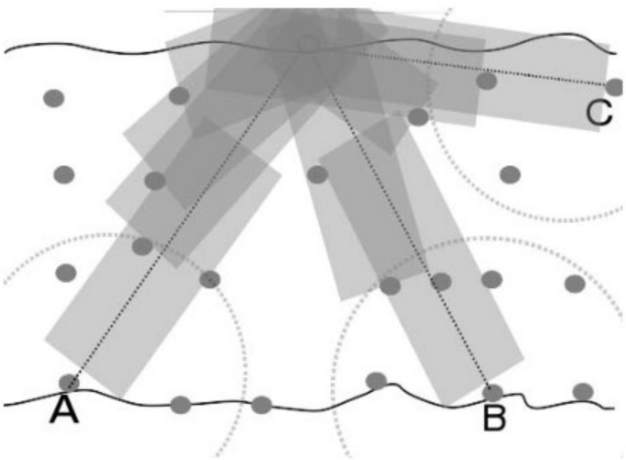


Figure 4.2 : HH-VBF [59]

The packet waiting delay and processing duplicate packets problem in [91] has been addressed in [92] . If the node qualifies as candidate forwarder then it saves the packet ID in the memory and during delay time it checks if the packet ID is already saved in the RAM so as to avoid duplicate packet sending in dense node network.

It is claimed that HH-VBF can avoid VOID areas by a feedback mechanism from the forwarder to its source, informing there is a void ahead, so that the source chooses another path. This area has to be refined in future but due to hop-by-hop nature, HH-VBF possesses much more signalling overhead as well as energy consumption compared to VBF and will be highly inefficient if a network configuration is constantly changing due to water current. Introducing a mechanism to ensure back-up paths in sparse network is needed. So that the optimum number of nodes which are needed to provide coverage to certain region should be deployed in a certain manner in order to guarantee reliability through sufficient back-up paths. These back-up routes can combat the dynamic nature of UWSN due to water current and node movement.

4.2.2 Focused Beam Routing (FBR)

FBR [60] is a scalable routing technique based on location information. It is suitable for both mobile and static underwater acoustic sensor network without the need of clock synchronization [93] . The idea of FBR is to restrain the flooding by forwarding the data packets to the candidate forwarding nodes within the transmitters cone θ as shown in Figure 4.3 and by controlling the transmission power so that the energy consumption is reduced.

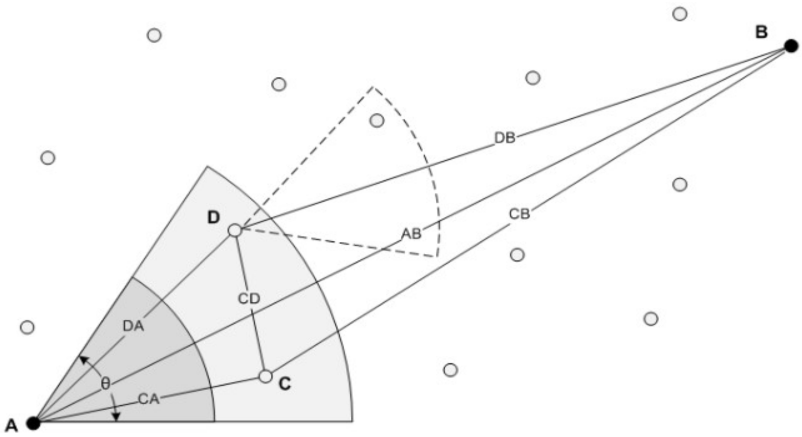


Figure 4.3: FBR, nodes within the transmitter's cone θ are the candidate forwarders [60]

In this routing protocol, the source node broadcasts CTS/Hello packet at a minimum transmission power P1 and receives it back from several nodes. Then it selects the one which is nearer to the sink as forwarder within the cone which originates from the source towards destination. In case it does not hear back the CTS packet it increases its transmission power level until it hears the CTS packet back. If it does not hear anything back it shifts the direction of its transmission cone to left or right [60] . In this way using different power levels, the packet is transmitted from a source towards destination as shown in Figure 4.3 [60] .

A discrete event underwater acoustic network simulator is used in [60] to assess the performance of FBR with different node densities and network loads. It was claimed that this technique is able to dynamically discover minimum energy routes with minimum network knowledge. Also it can avoid VOID at the cost of energy by increasing the transmission power of each node. Which will cost huge amount of energy on the overall network performance with randomly distributed sensor nodes.

There is a need to consider additional parameters in the cost metrics than the location information, in the selection process of forwarding candidate node, such as remaining battery power etc. In sparse node deployment regions, if the nodes are outside the transmission cone then there will be a huge energy overhead to reach a forwarding node. Similarly there will be a communication overhead due to the rebroadcast of CTS packet every time before transmission. And this situation will get worse due to random node deployment as there will be dense node concentration regions as well as sparse with the constantly changing network configuration.

4.2.3 Depth Based Routing (DBR)

DBR is a localization-free greedy-routing protocol which can take advantage of multiple sink network architecture. In DBR a node qualifies as forwarder if its depth is less than the transmitter's depth as shown in Figure 4.4 [61] . Depth information is acquired by depth sensor attached to the node. Each node maintains two queues: priority Q1 and packet history buffer Q2, in order to reduce the number of forwarding nodes and to control the duplicate transmission of the packet respectively. If a node qualifies as a forwarder then the node calculates holding time and inserts packet into the queue Q1, else the packet is dropped. The holding time depends on depth information. If the packet has been forwarded successfully its information is stored in the packet history buffer to suppress duplicate transmission [61].

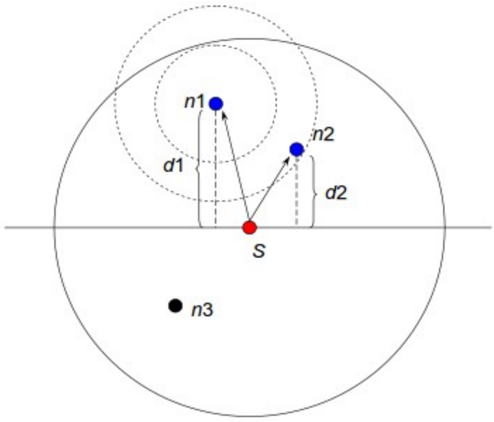
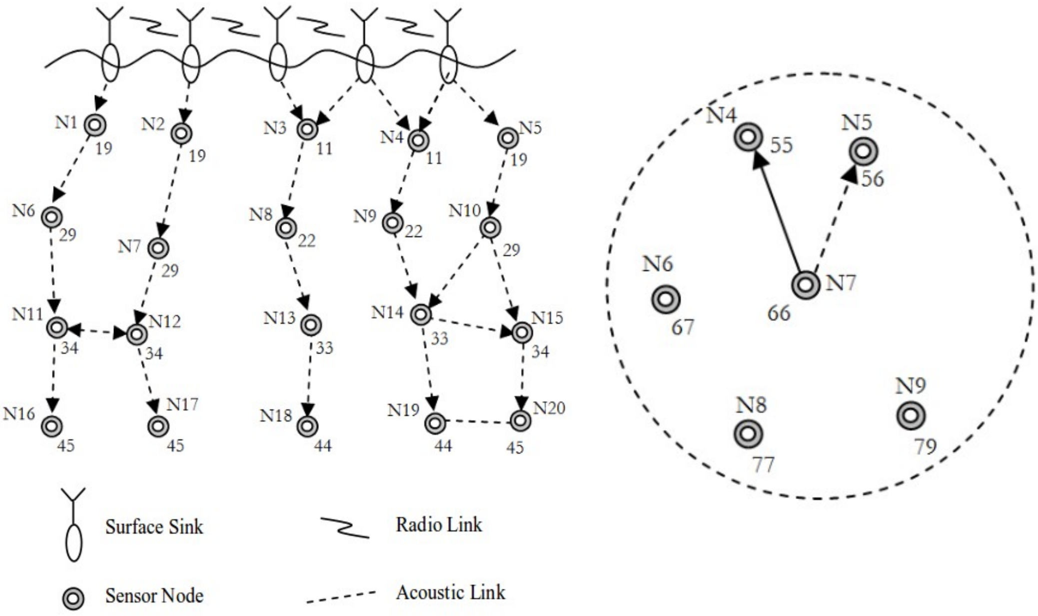


Figure 4.4: Forwarding Node Selection in DBR [61]

In [61] the performance of DBR was evaluated by considering packet delivery ratio, average end-to-end delay and total energy consumption using ns-2 simulator. The results show that DBR can achieve high packet delivery ratios (at least 95%) for dense networks with reasonable energy consumption. However for sparse networks, either a recovery algorithm needs to be explored or comprehensive node deployment strategy can combat the changing network configuration to avoid VOID areas where greedy strategies fail and provide back-up paths for reliable transmission of data.

4.2.4 Hop by Hop Dynamic Addressing Based Routing (H^2- DAB)

H^2-DAB [62] is a localization free routing protocol with no extra specialized hardware requirement and can take advantage of multiple-sink network architecture. In this routing protocol a unique ID is assigned to each sensor node with two digits as shown in Figure 4.5(a) [62] . The first digit denotes the distance from the sink in terms of the number of hops. The second digit denotes the distance from another sink in terms of the number of hops. Hence each node keeps a record of two sink nodes. The node with the smallest hop ID is selected as a forwarder as shown in Figure 4.5(b) [62] . The source node sends inquiry/hello packet to next layer nodes (above itself); if no reply, waits, retries, if again no reply forwards it in the same layer to a node having connection to upper layer. For next interval of data transmission new hop IDs are assigned, so it is a node mobility tolerable scheme [62] .



(a) (b)

Figure 4.5: H^2-DAB (a) Assigning hopIDs with the help of Hello packets, (b) Selecting the next forwarding node [62]

The performance of H^2-DAB was evaluated in terms of the delivery ratio and the end to end delay using ns-2 in [62] . It was observed that it can achieve high delivery ratios (at least 90%) in both dense and sparse networks with minimal delay and energy consumption. However, the extension of this algorithm to avoid VOID and support changing network configuration has to be worked. Also the basic multi-hop routing problem in which nodes near the sink drain more battery remains an issue.

4.2.5 Directional Flooding based Routing (DFR)

DFR is a localization-based, controlled flooding technique. In this routing protocol, the forwarding angle (<SFD), as shown in Figure 4.6 [94] increases for each forwarding node, from the source to the destination. For a node to qualify as a forwarder, its forwarding angle should be greater than previous forwarder's forwarding angle and its distance to the sink should be less than the previous forwarder [94] .

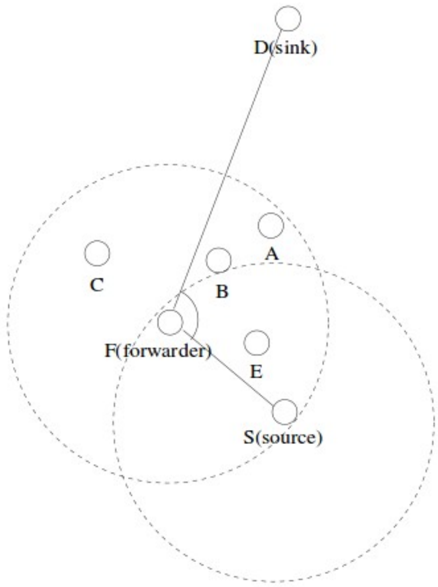


Figure 4.6: Packet transmission in DFR [94]

The performance of DFR was evaluated against VBF with the ns-2 simulator. It offers better delivery ratio, less communication overheads and shorter end-to-end delay. However, in areas where link quality is not good, multiple nodes can forward the same packet consuming the network resources and there is no feedback path mechanism suggested to avoid VOID.

4.2.6 Sector Based Routing with Destination Location Prediction (SBR-DLP)

SBR-DLP [63] is a partial localization based routing protocol in which a node knows its own location and predicts the location of the destination node which is assumed to be mobile. In this routing protocol a sector based flooding is performed. Prior to the data transmission, the source node S with a pending packet broadcasts a check neighbour (*chk_ngb)* packet containing its current location. The nodes closer to the sink node send a reply packet. Then node S starts to label and locate the first sector ensuring that the sector is bisected by a vector SD, a virtual line from source S towards destination D, as shown in Figure 4.7 [63] . The forwarding node is selected within the first sector bisected by vector SD and close to destination. The remaining sectors are prioritized and labelled according to the angular differences from vector SD [63] .

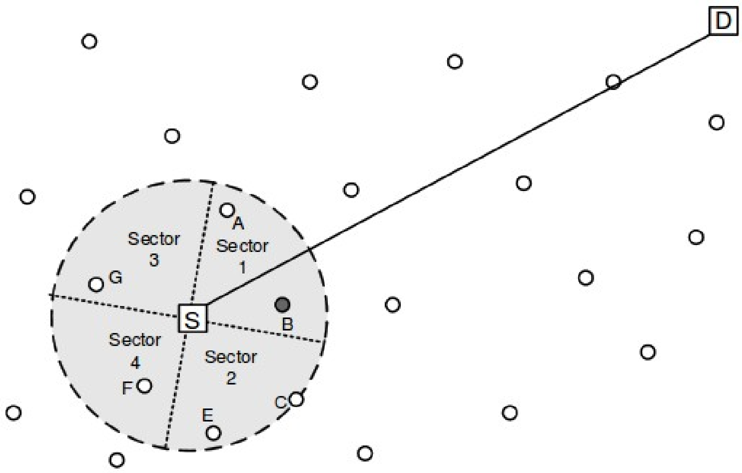


Figure 4.7: Forwarding node selection in SBR-DLP [63]

In this routing protocol the destination location prediction of the mobile sink is achieved. Whenever a mobile sink deviates from its scheduled movement trajectory, it broadcasts a notification packet (NTF) to its one hop away neighbours. Node mobility is handled as the source node filters out the list and removes the nodes which might go out of range before acknowledging the data packet, by considering the propagation delay and the velocity of the nodes.

It is shown through simulations by the authors that a simple location prediction mechanism could help improve the packet delivery ratio significantly. However, if the destination node does not follow the pre-planned path due to water currents, this will be an issue. SBR-DLP is highly adaptive to network dynamics such as nodes joining and leaving the network, because of its reactive hop-by-hop packet routing mechanism. However this will be a cause for high energy overhead. Location prediction mechanism of a node, due to water current can be highly unpredictable. Water movement in the oceans is uncertain and can take extensive mathematical modelling to predict anything closer to reality.

4.3 Multipath Based Protocols

In multipath protocols more than one path is established from the source towards the destination. It improves reliability, robustness and packet delivery ratio by forwarding the packets on multiple paths simultaneously and avoids a bottleneck being formed around the Sink. However, duplicate transmission and computation involved in maintaining more paths remains an issue to be dealt with efficiently. Some of the multipath based protocols worth mentioning are:

(i) Multipath Virtual Sink Architecture

(ii) Resilient Routing Algorithm

(iii) Reliable and Energy Balanced Routing Algorithm

4.3.1 Multipath Virtual Sink Architecture (MVSA)

This topology [65] ensures reliability due to a lot of redundancy and claims to provide many communication paths so that it continues to function even when a significant portion of the network is non- operational (due to VOID or temporary loss of connectivity). It actually does simultaneous retransmissions instead of sequential retransmission, ensuring lower delay.

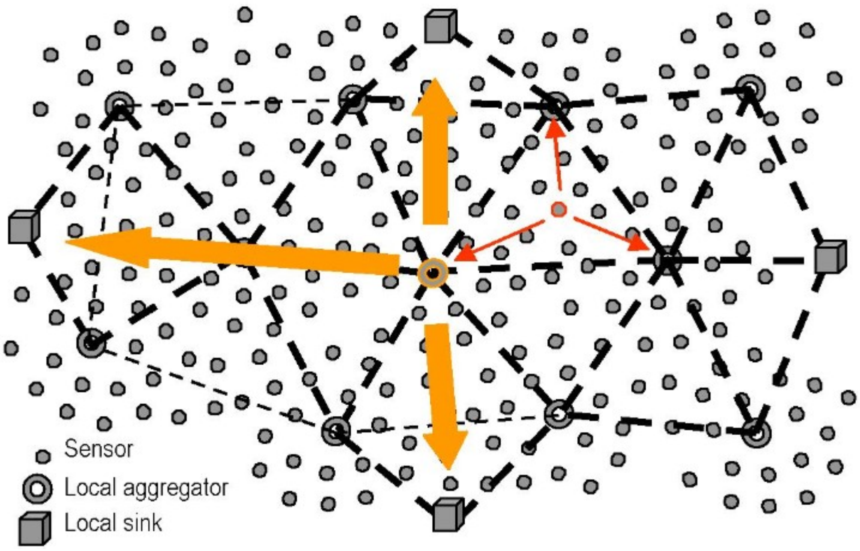


Figure 4.8: Multipath Virtual Sink Network Architecture [65]

As shown in Figure 4.8 [65] the network consists of clusters of sensing nodes with one or more reachable aggregators. These aggregators form a mesh network which further connects local sinks. 2-tier topology is shown in the figure; the number of tiers is flexible and can be adapted to meet deployment requirement. The local sinks are connected via high speed RF link or optical fibre (undersea). So it is ensured that data is delivered from sensor node to one or more of these local sinks which form virtual sink architecture. High congestion is avoided by using multiple paths towards multiple sinks and no retransmission on the same path [65]

The simulation results in [65] using Qualnet simulator showed an overall better network performance in terms of Packet Delivery Ratio and average end-to-end delays as compared with single-path single-sink architecture with a finite number of permissible retransmissions. However it is not an optimal or reliable design as far as number of sensor nodes, energy consumption and cost are concerned. Due to changing network configuration due to node movement, it will cost energy as well as communication overhead to maintain these clusters for communication as RF links underwater are not feasible (as discussed in chapter 3). Similarly huge network of optical fibre to connect the virtual sinks is not an optimal design for large-scale UWSN which is normally the case, as small scale applications are not the kind of issues to be solved by UWSN.

4.3.2 Resilient Routing Algorithm (RRA)

In this routing algorithm [66] , winch-based sensor nodes are anchored to the sea bottom and fitted with a buoy as shown in Figure 4.9. Buoy pulls the sensor nodes to ocean surface. The depth of each sensor node can be regulated by adjusting the length of the wire that connects the sensor node to the anchor, by means of an electronically controlled engine that resides on the sensor node [66] . There is a centralized network manager and multi-hop connections are established between each source and sink. The routing solution works in two *phases;* Phase 1: The network manager assigns optimal primary and backup multi-hop paths using integer linear program (ILP) in order to minimize the energy consumption of the *nodes.* Phase 2: An online distributed solution guarantees connectivity of the network (by locally repairing the paths or by switching the data to a backup path).

In the simulation results, the authors show the effectiveness of RRA in providing energy-efficient data paths and handling the node failures. This architecture is claimed to have certain strengths e.g. the sensor nodes are not vulnerable to weather, they do not obstruct the ship navigation and cannot be detected by intruders etc. However this routing scheme is proposed for long term applications and with the use of this architecture and winch-based anchored nodes, the overall cost will be a major issue to deal with for larger monitoring regions.

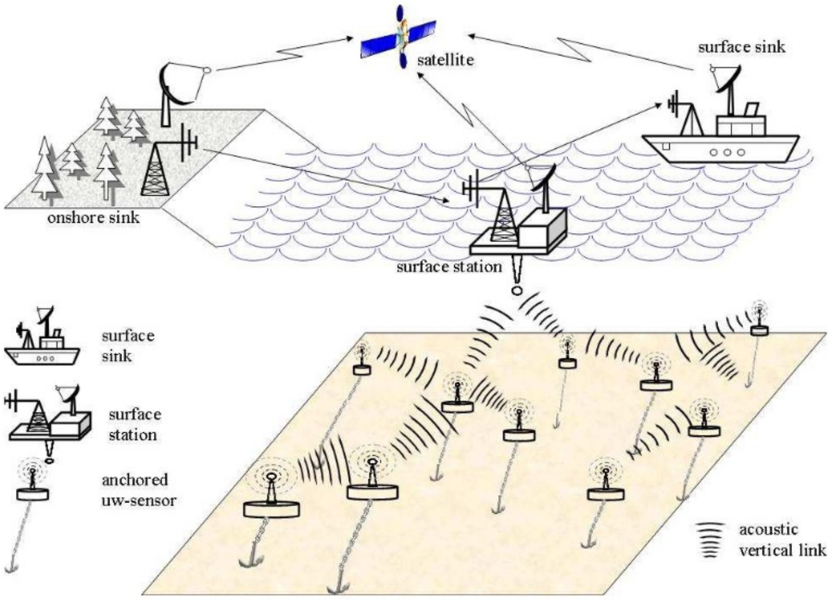


Figure 4.9: Proposed 3D UWSN architecture in RRA [66]

4.3.3 Reliable and Energy Balanced Routing Algorithm (REBAR)

In REBAR [67] , the authors introduce an energy model as shown in Figure 4.10 . This energy model is based on the concept that the sensor nodes near the sink drain their battery quicker than those which are farther as explained below. In this architecture, the monitored space is divided into adjacent spheres (semi-spheres) as illustrated in Figure 4.10. Sensor nodes are randomly deployed and sink is stationary and fixed at the centre of the monitored region's surface. Each layer, namely S1, S2, S3, S4 and S5 is of the same thickness.

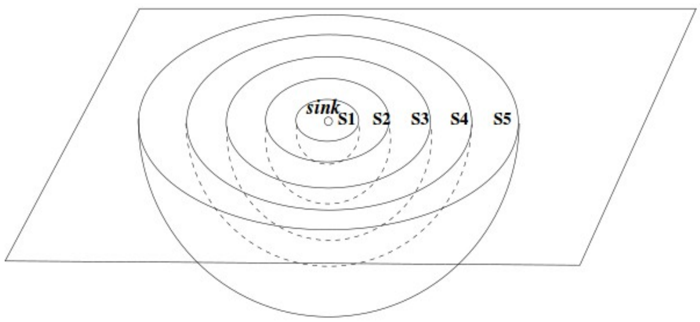
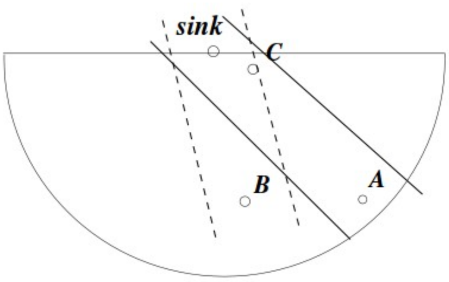
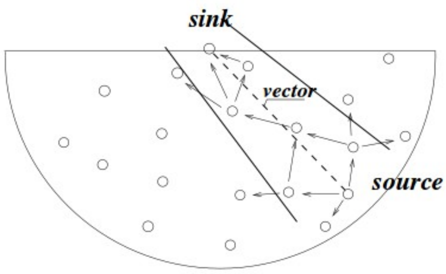


Figure 4.10 : The sphere energy model in REBAR [67]

The sensor nodes in the outer most layer S5, only need to forward the data packets generated by themselves. The sensor nodes in layer S4 will forward their own as well as layer S5 nodes data packets, towards the inner layer S3. Similarly, S3 will forward its own data packets as well as data packets from layer S4 and S5 to layer S2 and so on. This implies that if a 3D sensor network is divided into layers of a sphere with the sink in the middle, then each layer closer to the sink will be responsible to forward its own data as well as data from its outer layers. Since data forwarding consumes energy, the sensor nodes in the inner layer, will consume more energy than its outer layer. Hence the semi-spherical layers (S1, S2, S3, S4 and S5) around the sink denote the different levels of energy consumption by the nodes present in those layers. For example the overall energy consumption of the nodes in layer S1 will be more than the energy consumption by the nodes in any of its outer layers i.e. S2, S3, S4 or S5, as layer S1 is nearer to the sink than those layers. Based on this energy model a routing algorithm is proposed. The concept is a bit similar to VBF, vector based forwarding routing protocol [91] and FBR, focused beam routing protocol [60] . In REBAR multipath routing is achieved as shown in Figure 4.11.



(a)



(b)

Figure 4.11: a) The constant constrained radius, (b) Routing process in REBAR [67] .

A vector is formed from the source to the sink and the packet is routed within the conical pipe in the direction towards sink. The idea is that by reducing the diameter of the pipe nearer to sink, it will involve less nodes in nearer layers to qualify as forwarders. Hence less nodes will drain their battery. In REBAR the node movement is considered positive in a way to distribute the energy consumption in the network. When the nodes move from their position they might not be involved in the next forwarding process and some other node might take up the role. A mechanism to avoid VOID is also introduced but it does not seem very efficient. VOID will be detected by using one of the methods as described in [95] or [96] by the node at the boundary of a VOID. This node, located at the boundary of the Void will forward the packet by flooding it to all the nodes around it rather than routing it through the conical pipe (having void) towards the sink. Hence data packet will be forwarded to the sink by avoiding the VOID. Although these routing concepts are very appealing but node movement will charge them high communication as well as energy overhead in order to reconfigure the dynamic 3D nature of the network after node movement or failure. This topped up with random distribution makes the whole set-up unreliable in terms of optimum use of resources. With random distribution there will be dense node regions as well as sparse and VOID creation will always be an issue to deal with costing huge energy resources.

4.4 Clustering Based Protocols

In clustering based protocols sensors are grouped together to form clusters. Clustering approach has two types of nodes; cluster head node and cluster member node. Cluster head node collects data from all the cluster member nodes and transmits it further to destination. Some of the cluster based protocols for UWSNs are:

(i) Distributed Underwater Clustering Scheme

(ii) Pressure/Hydrocast Routing

(iii) Dynamic Cluster Based Routing

4.4.1 Distributed Underwater Clustering Scheme (DUCS)

DUCS [69] is a distributed energy-aware routing protocol. It is localization free scheme and minimizes proactive routing overheads and uses data aggregation. It addresses random node mobility and compensates high propagation delays by a continually adjusted timing advance combined with guard time to minimize data loss. Combining DUCS with TDMA/CDMA reduces interference and improves communication quality [69].

DUCS protocol works in rounds. Each round has two phases i.e. Setup/Cluster formation phase and Steady state/Data transfer phase. In the setup phase each node has a probability to become a cluster head. Certain number of cluster heads are elected by cluster head selection algorithm depending on their residual energies. Once elected as cluster head the nearby non-cluster head nodes send join request message to form a cluster as shown in Figure 4.12 [69]. The cluster head upon receiving join requests computes a TDMA schedule for cluster members depending on propagation delay . After each round before cluster formation new cluster head is elected depending on residual energy of cluster members.

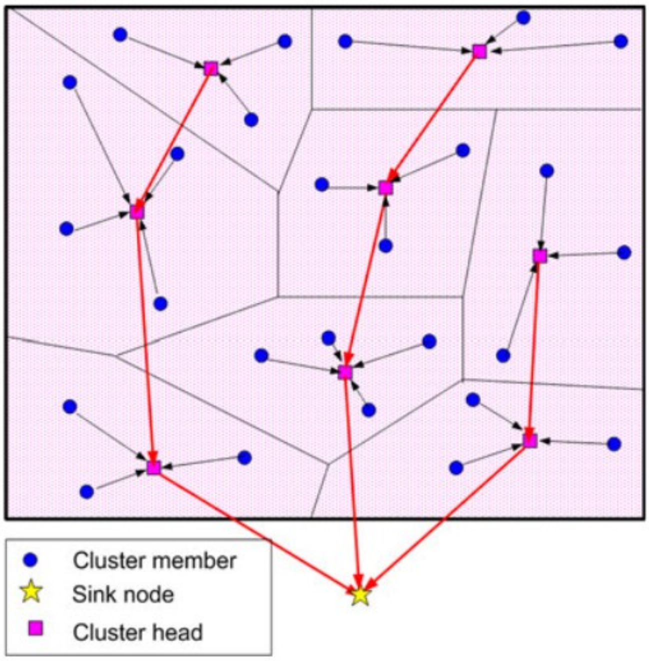


Figure 4.12: Clustering in DUCS [69]

After the transmission of a frame to the cluster head, one time slot is used for cluster maintenance, by cluster maintenance algorithm as shown in the time line of DUCS in Figure 4.13 [69] Due to node mobility, the node's positions vary with time, and it is necessary to modify the cluster members and TDMA schedules accordingly. After receiving data from cluster members, cluster head performs signal processing to compress data together with its own data and forms a composite signal which is delivered to the sink through other cluster heads using CDMA and multi-hop routing [69].

During the cluster maintenance time slot each node computes the propagation delay towards its cluster head node using the ToA (Time of arrival) technique. If the propagation delay of a node differs more than 50% of the initial propagation delay, that node re-estimates its distances towards each cluster head node. All the cluster head nodes are taken into account. In this way, again the nearest cluster head node is selected.

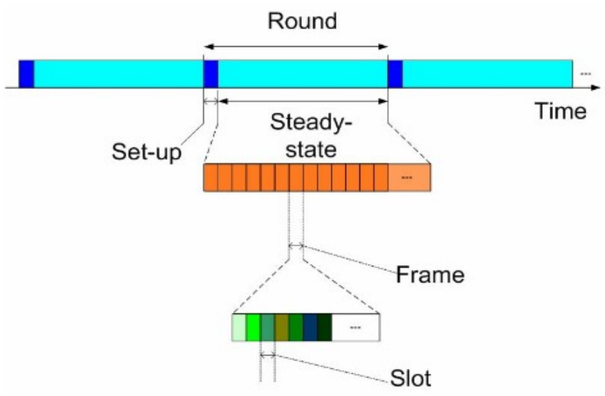


Figure 4.13: Time line of DUCS [69]

The simulation results carried out in [69] using ns-2 showed that DUCS achieves high packet delivery ratio while considerably reducing the network overhead. However as each node can become a cluster head and performs data collection and aggregation, this will result in expensive and complicated hardware. Frequent division of sectors in the set-up phase will consume the network resources. Also energy overhead in cluster formation keeping in mind the changing network configuration and sparse network remains an issue. During cluster maintenance time slot if a node re-estimates its distance from each cluster head node, this will also cause energy overhead.

4.4.2 Pressure/Hydrocast Routing

This is a depth/hydraulic pressure based anycast routing protocol. In this protocol [70] a cluster of nodes (called *forwarding set*) is selected based on the maximum progress of the nodes towards the destination. The maximum progress of the nodes is computed based on the packet delivery probability and the closeness of the nodes to the destination. The source node selects a maximum progress node towards the destination. Then the nodes, existing in the area that covers half of the range of the selected maximum progress node, are selected to form a cluster. The information about the selected cluster member nodes is then appended to the data packet.

In order to suppress the transmission of redundant data packets, a priority based approach is used. A node with the maximum progress has the highest priority and the shortest time-out timer for the transmission. All other nodes in the cluster suppress their transmission upon receiving the transmission of the high priority node [70] . If a node itself is at the minimum depth level among its neighbours then a local maximum recovery mechanism is introduced as shown in Figure 4.14 [70] . The node with local maximum state performs limited flooding; only the nodes existing on local maximum surface area participate in the flooding process. The nodes existing on the surface area are determined using a tetrahedralization method. The tetrahedralization method states that, a non-surface node is a node which is surrounded by its neighbours; otherwise the node is a surface node. In this way, after finding the surface nodes the packet is transmitted from one surface node to the other surface node and so on. After a few iterations the data packet is routed to a node where the greedy mode is restored. For instance, in Figure 4.14 there are two local maxima, LM1 and LM2. LM1 maintains a path to LM2 which has a path to node S. A packet can be routed from LM1 to LM2 to S and then the greedy mode is restored and the packet can be delivered to the node on the ocean surface [70] .

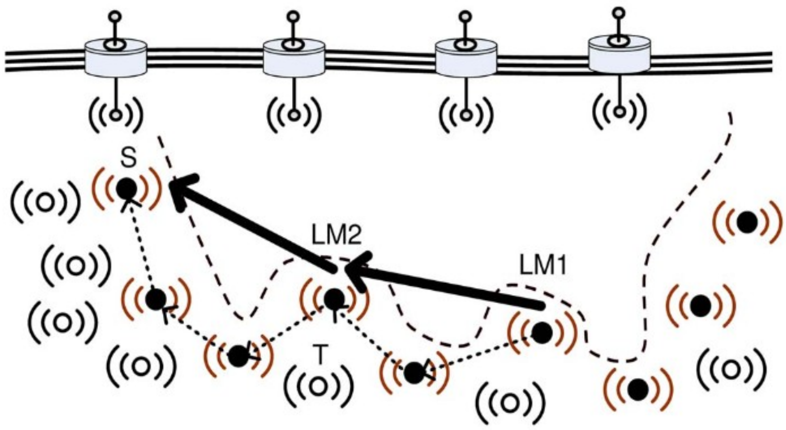


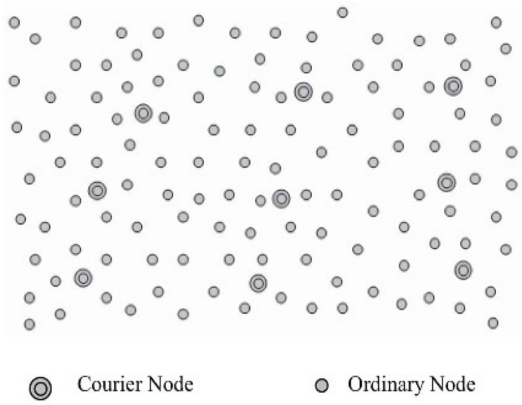
Figure 4.14: Recovery mode in Hydrocast routing [70]

The simulation results in [70] using Qualnet simulator were compared with DBR protocol and VOID area problem was overcome. It was shown that DBR provides higher delivery ratio with low node density as there is no packet suppression mechanism to control repetitive multi path transmissions. However, with recovery support Hydrocast outperforms DBR in terms of delivery ratio, end to end delay and reliability. However, multiple copies of data delivered to sink will be a burden on the network resources. And energy overhead due to pressure sensor operation has not been discussed.

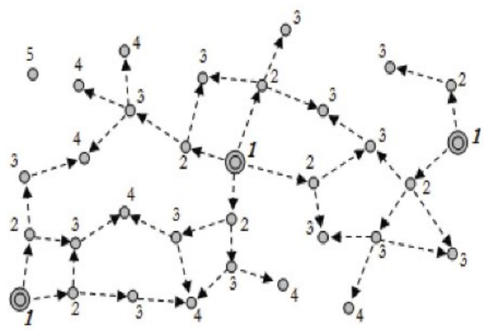
4.4.3 Dynamic Cluster Based Routing (DCB)

DCB [71] is based on multi-sink architecture. There are ordinary nodes and courier nodes in this architecture as shown in Figure 4.15(a). The ordinary nodes are deployed from the water surface to seabed at different depth levels with buoyancy control. The courier nodes are fitted with a piston module which helps to push the node inside the water at different defined depth levels and then pull back to ocean surface by positive and negative buoyancy. Courier nodes reach different defined depth levels, stop for specific time and receive data packets from the ordinary nodes [71]. Then they travel towards the surface sinks to deliver the data packets.

In this multi-sink architecture, if data packets reach one of the sinks, they are considered delivered successfully. The Surface buoys (sinks) are fitted with radio and acoustic modems. These multi-sinks at the surface use RF modem to communicate with each other and courier nodes and acoustic modem to communicate with other sensor nodes [71]. Hop Ids are assigned as shown in Figure 4.15(b) [71].



(a)



(b)

Figure 4.15: DCB, (a) Node deployment, (b) Assigning the hopID's [71]

The performance of this routing protocol was evaluated in [71] using ns-2 and it is claimed that DCB provides a reliability with less energy consumption. Also it handles node mobility and provides wider coverage. However, there will be an end-to-end delay in the transmission due to courier node's physical movement towards the sink to deliver the data packets. In addition to this, the ordinary nodes have to wait for a certain time for the courier node if they received data, just after the courier node moved away to travel to wards the sink to deliver data, till the next time it will arrive.

4.5 Miscellaneous

4.5.1 Adaptive Routing

Adaptive routing [97] is localization based scheme. The key idea of this routing scheme uses a packet priority based network resource allocation. Hence an important packet will be delivered reliably with the least end-to-end delay compared to an ordinary packet. The packets are classified as ordinary, intermediate and emergent and are assigned priority in the range of [1, 100], where 100 means the highest priority. Underwater sensors are deployed at different depths using buoyancy control, they can move freely in the horizontal plane but can deviate slightly in the vertical plane. In this way sensor nodes at same depth form a layer.

The main tasks performed at each node are: neighbour discovery, priority calculation and routing decision. The neighbour discovery is done by periodically broadcasting the HELLO packets with node location, ID and buffer size. Furthermore, acknowledgement packets (ACKs) based on epidemic routing are broadcasted back from sink to suppress redundant packets existing in the network. And each node keeps all these ACKs in order to decide whether to keep or drop whenever a packet is received. The packet priority is calculated based on emergency level, age and node status such as residual battery and the density of neighbourhood [97].

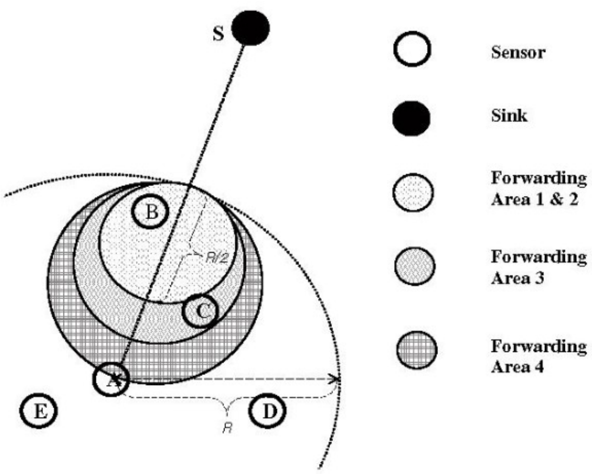


Figure 4.16: Illustration of routing decision procedure [97]

There are four routing states (1, 2, 3, 4) introduced for four priority ranges [0, 25], [25, 50], [50, 75], [75, 100] respectively. These states indicate the number of copies and are mapped to corresponding forwarding areas. For example, if a packet has priority 45 then it will fall in the state 2 according to priority ranges and two copies of this packet will be forwarded in the forwarding area for 2 (which is the same for 1) as shown in Figure 4.16 [97]. The diameter for forwarding area for states 1, 2, 3 and 4 is set to R/2, R/2, 3R/4 and R respectively. In general, a higher priority of a packet corresponds to larger forwarding area.

The performance of Adaptive routing protocol was evaluated through simulations in [97] and compared to Epidemic routing protocol and Single copy routing protocol. It was shown that it provides a better overall performance in terms of delivery ratio, energy consumption and average delay. However, the effect of node mobility due to water current and VOID region formation due to random node deployment and changing network configuration, stays a problem. Also keeping all the ACKs packets in the memory will cause a huge overhead for large network in terms of node memory which can be avoided by putting a timer for the ACKs packets to stay in the memory.

4.6 Summary

In the last few years, much research work has been done on the networking layer of UWSNs. There are a number of routing protocols being developed. Some of them claim to consider changing network configuration in the future work. The flooding based routing approaches for UWSN (such as VBF [91] , FBR [60] , DBR [61] , H^2-DAB [62] , SBR-DLP [63] etc. are simple at computation and have acceptable end-to-end delay, low processing overhead and can be utilized for delay sensitive applications [66]. However, the energy consumption in duplicate transmission, congestion and channel sharing remains a problem. This has been tried to be overcome by some specific individual approaches by each protocol to some extent. However the existence of VOID regions (a region which is created due to random deployment, node movement or failure and is not covered by any other node) and changing network configuration limits this approach. Some of flooding based approaches involve localization, another energy overhead, as the network configuration is constantly changing with water current. Hence localization is an issue in itself for UWSNs.

The protocols utilizing multi-path transmission (such as MVSA [65], RRA [66] and REBAR [67] etc. as discussed in this chapter) compared to flooding based approaches offers acceptable reliability, energy efficiency, low end-to-end delays, less congestion and interference but more computation involved in maintaining more paths [68]. Therefore efficient solutions are needed to avoid repetitive transmission. In addition the changing network configuration has been completely ignored and this will cause loss of connectivity in randomly deployed sparse network.

The clustering based protocols (such as DUCS [69], Pressure Routing [70] and DCB [71] etc.) are complex in terms of processing overhead due to node mobility and re-clustering. An efficient routing mechanism for re-clustering can reduce these overheads and this area needs to be worked further in order to avoid loss of connectivity due to changing network configuration and random node deployment.

Adaptive routing [97] , localization based scheme seems efficient for static networks. The key idea of this routing scheme is packet priority based network resource allocation. Hence an important packet will be delivered reliably with least end-to-end delay compared to an ordinary packet.

Considering the unique nature of UWSN (i.e. high propagation delay, high error probability, low bandwidth capacity, 3D environment and dynamic configuration due to node movement) evaluating the performance of these routing protocols by random node deployment is highly unreasonable. Although it is a common practice as adapted from terrestrial WSN (which are effectively 2D networks). And due to the reason that an effective 3D node deployment can be highly complicated. To rectify this issue we present a 3D node deployment strategy in Chapter 5 to improve routing performance.

CHAPTER 5

Improving the Routing Performance by 3D Node Deployment Strategy

5.1 Introduction

In this research work we focus on improving the networking protocol performance by testing a number of strategies for node deployment in an underwater environment. Most of the networking protocol design concepts and node deployment strategies for UWSNs are adapted from terrestrial WSNs, and re-designed with some changes to fit the underwater environment. However, due to the fundamental differences between UWSN and WSN technology (as discussed in Chapter 2), these designs are not suitable for a complex 3D underwater environment. Although terrestrial WSNs mostly operate in 3D but the third dimension, which is the height is significantly small compared with the other two dimensions (length and width of the network), hence the two-dimensional (2D) assumption for terrestrial WSN does not lead to any major inaccuracies. However, a 3D network design is much more complicated than a 2D counterpart. Most of the networking protocols developed for UWSNs are tested in 2D with random node deployment and results assumed to be acceptable to random node deployment methods in 3D, which are simple to implement in the simulation tools. However, keeping in mind the size, sensing range and communication range of underwater sensor node, this causes major waste of resources. Unlike tiny WSN nodes, UWSN nodes are big in size and with random deployment of such nodes, it is not feasible to have optimum use of these nodes for full coverage and connectivity. The strategies proposed in this research work have been a result of careful consideration of the unique nature of underwater acoustic communication, such as high and variable propagation delay, low bandwidth capacity, high error probability and temporary loss of connectivity. In addition the specific characteristics of UWSN which makes it different from terrestrial WSN, such as 3D architecture and loss of connectivity due to node mobility etc. have also been taken into account.

5.2 Node Deployment in UWSNs

A large number of algorithms have been proposed recently for node deployment in 3D environments as discussed in [98]. In one of them [99], the authors present a node deployment strategy with 100% sensing coverage of a 3D volume with minimum number of sensor nodes. A metric called volumetric quotient is defined which is the ratio of volume of polyhedron to the volume of its circumsphere, the higher the volumetric quotient the smaller the number of sensor nodes required for full 3D sensing coverage. Further in this paper [99] several space filling polyhedrons deployment such as cube deployment, hexagonal prism deployment, rhombic dodecahedron deployment and truncated octahedron deployment are analysed. It is shown that the truncated octahedron deployment is the best strategy for full 3D sensing coverage of the desired volume with minimum number of nodes as shown in Figure 5.1. It has the highest volumetric quotient and results in minimum number of nodes for full coverage. If the sensor nodes are placed in the middle of each cell in the 3D lattice of truncated octahedron as shown in Figure 5.2, each point in the monitored volume will be covered by the sensing range of one node.

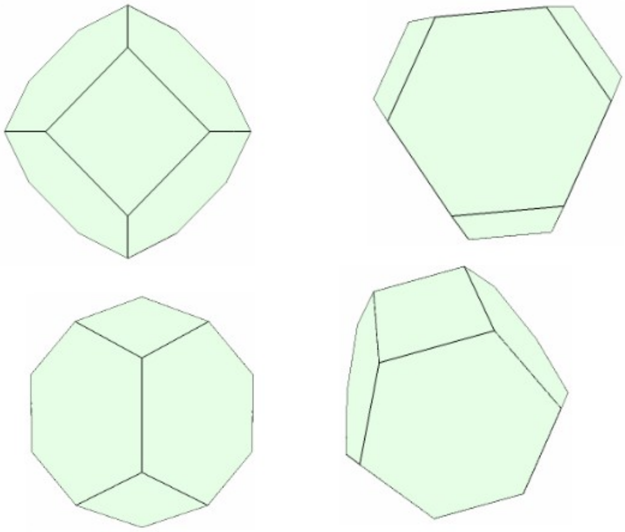


Figure 5.1 : 3D Truncated Octahedron from different angles [99]

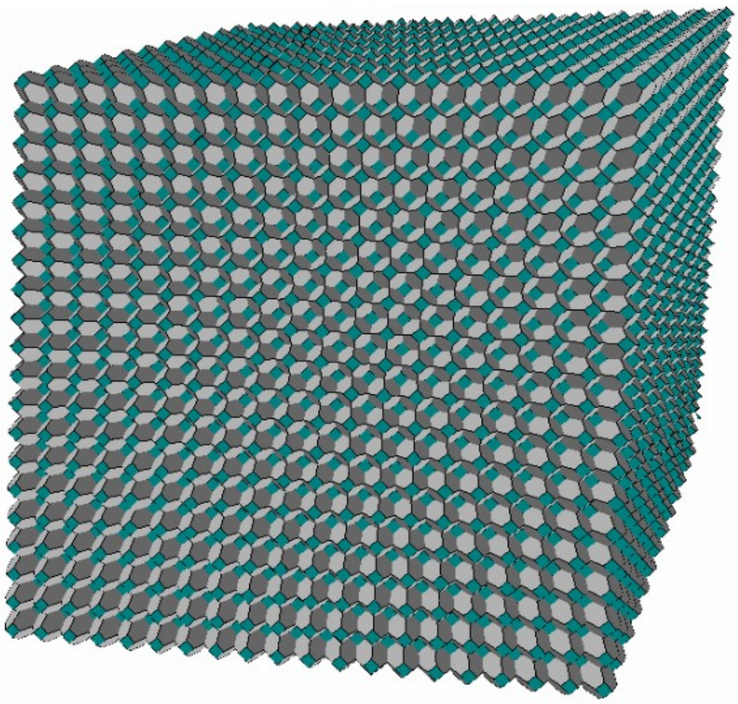


Figure 5.2: A three-dimensional network volume consisting of 20×20×20 nodes when placement strategy is truncated octahedron [99]

There can be applications of sensor networks in which each point in the monitored space, needs to be covered under the sensing range of one node however because of the constantly changing network configuration and challenging wireless acoustic connectivity in UWSN, most of the applications need better and more reliable coverage both in terms of sensing and communication. For instance, if a node dies, moves or a connection fails, another redundant node should be able to continue to provide the full coverage. On the other hand a dense network, with sensor nodes having several reachable neighbours could increase networking problems due to interference and collisions. Hence the scheduling of communication tasks can become extremely challenging. In random deployment this could be the case in some regions of the network whilst other regions might be sparse with no back-up routes or not being fully covered. We propose a simple (to implement) yet novel and reliable 3D node deployment strategy with fewer but highly effective redundant nodes which significantly increase the intensity of the coverage and reliability of the network.

Underwater sensor network is a 3D network unlike terrestrial wireless sensor network. The flow of information generally needs to be from seabed to sea top and not along a horizontal path as in terrestrial wireless sensor network. We investigated the routing protocols in detail for UWSN, and it is revealed that many routing strategies are developed for UWSN, to discover an efficient route between source and sink, as discussed in Chapter 4. However most of the concepts behind these routing protocols have been adopted from terrestrial WSN and redesigned with minor changes to fit in underwater environment. Similarly these protocols are tested using the same methods as for terrestrial WSN. For example most of them are tested in a 2D set-up or using random deployment strategies for 3D set-up. The results based on 2D deployment cannot be inferred for practical 3D UWSN scenarios. In addition the results taken from random deployment in 3D will not guarantee the optimum use of resources and reliability, which the nodes can offer to the network if deployed properly in a certain manner. There is significant room for improvement and further development in these routing protocols in order to ensure reliability through efficient use of resources and to increase the network life time by intelligently distributing the network traffic load. What is needed is to improve the performance of these routing protocols for the 3D underwater environment, by intelligently deploying the sensor nodes in a simple yet effective manner, so that the reliability increases with the efficient use of resources and the water current has minimum impact on the node movement and network performance.

Underwater wireless acoustic propagation is characterized by high propagation delay, high error probability and limited bandwidth capacity. Further it suffers from Dopplers effect, spreading loss, scattering loss, path loss and absorption. The signal is also subject to multipath propagation through a channel whose characteristics vary with time and are highly dependent on the location of the transmitter and receiver [1]. One of the major difference of acoustic communication with respect to terrestrial radio communication is the strong anisotropic nature of acoustic signal propagation whereby horizontal channels are harsher than vertical channels [100]. The multipath structure depends on the link configuration, which is primarily designated as vertical or horizontal, while vertical channels exhibit little multipath, horizontal channels may have extremely long multipath spreads. Most notable in the long- and medium-range channels, multipath propagation causes severe degradation of the acoustic communication signals [1] [38]. Combating the underwater multipath to achieve a high data throughput is without exception considered to be the most challenging task of an underwater acoustic communication system [1]. This forms the basis of our node deployment strategy.

5.3 Proposed Approach

We propose a node deployment strategy for 2D and 3D underwater sensor network scenarios to improve the routing efficiency in terms of energy and end-to-end delay. Our design ensures a full reliability to the data communication hence there will be no packet loss. We present several test scenarios beginning from the basic one path connectivity to providing several back-up paths for reliable communication at no extra cost of network resources compared to the essential one path connectivity. It is observed through several simulation results that our proposed 3D node deployment strategy improves the overall routing performance of the network. It gives much better results in terms of end-to-end delay and energy consumption, compared to random 3D node deployment strategy.

In our proposed configuration, the main direction of communication is vertical from sea bed to sea top unlike terrestrial WSN adopted designs which normally consider horizontal direction of communication. In sea, the sensed data from under the sea is required to travel to the base station which is located at the sea top. Hence the preferred path of data flow is vertical from sea bottom to sea top as shown in Figure 5.3 below.

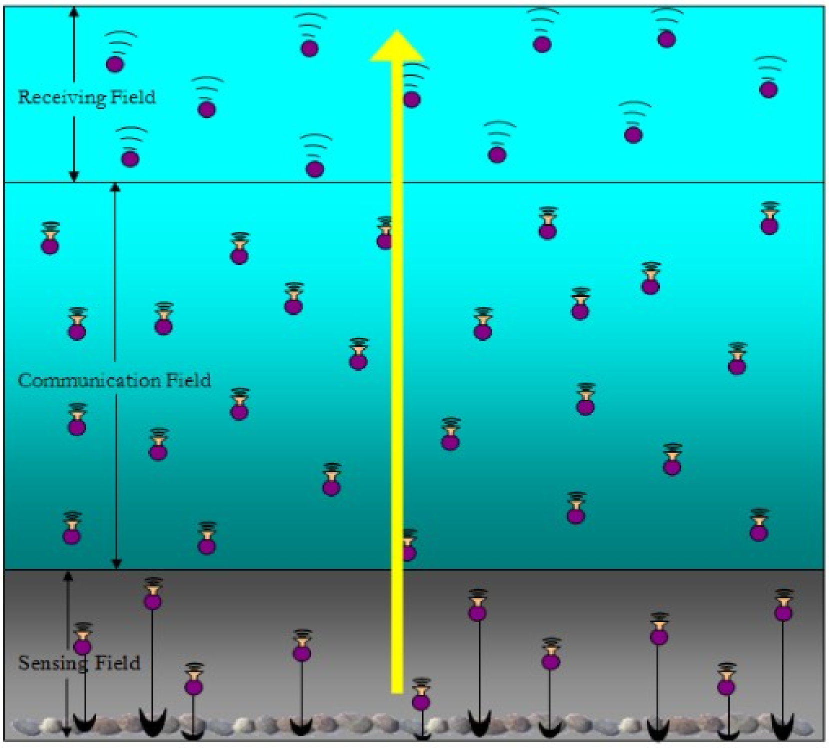


Figure 5.3: Layers and path of communication in UWSN

To simplify our UWSN design, we divide the subsea into three layers; Sensing Field, Communication Field and Receiving Field as shown in Figure 5.3. The demarcation of these layers is arbitrary and conceptual. They can overlap for certain design conditions depending on application requirement. Although all of the sensor nodes in the UWSN are capable of sensing data but in the Sensing field, we are emphasizing the fact that this layer of UWSN will be least involved in relaying a data packet as being at the lowest depth, for example at the seabed. Hence its main task is sensing and forwarding its own data. The Sensing field can be comprised of anchored or mounted, known location based sensor nodes. In this way it can provide the backbone to the network architecture for localization purpose. The Communication Field is the main field (regarding routing/ forwarding data) located in the middle of the Sensing and Receiving Field. It is comprised of mobile sensor nodes. The Receiving Field consists of destination sinks and mobile shallow water sensor nodes. These nodes can be electromagnetic sensor nodes so as to benefit from high bandwidth of electromagnetic communication in shallow water. Mainly we work in the Communication field to propose an effective 3D node deployment configuration to improve the networking performance.

5.4 Our Design Strategy

For our simulation experiments we use Vector Based Forwarding (VBF) [91] as a baseline routing protocol. It is a well-known routing protocol in the UWSN research community (as discussed in Chapter 4). In VBF for the data packet delivery a virtual pipe around the vector from the source node to destination node is formed and the packet is forwarded to the nodes within that virtual pipe. We develop several mechanisms to improve its performance step by step from single line of communication to adding back-up paths in 2D and finally to our effective 3D node deployment strategy. These steps include,

- vertical line of communication to minimize multipath effect.
- 2D node deployment strategy
- 3D node deployment strategy
- random 3D node deployment strategy (to compare our 3D deployment strategy results)

We used Aqua-Sim as a simulator for experimenting. It is an extension of the popular ns-2 simulator which is extensively used among the networking research community. We deploy a variable number of sensor nodes in a variety of 2D and 3D configurations to observe the network performance step by step and finally propose a 3D node deployment strategy for efficient network performance. For our simulation experiments we consider an underwater volume of $500 \times 500 \times 500 m^3$ and deploy variable number of sensor nodes. Each of these sensor nodes have 10000 joules initial energy, 2 watts transmit power, 0.75 watts receive power and 0.008 idle power consumption. Each sensor node is equipped with omnidirectional antenna of 100m range with spherical spreading. The data packet size is 50 bytes and the bit rate is 10kbps.

5.4.1 Vertical Line of Communication to Minimize Multipath Effect (Test I & Test II)

In our first simulation test set-up (Test I), we deploy 6 sensor nodes at a distance of 100m from each other in a vertical line from seabed to sea top shown in Figure 5.4. There is one source node, one sink node and four forwarding nodes involved in forwarding the 50 sensed data packets from source node to sink. The node placement is 100m apart in the vertical direction on the y-axis (at 0, 100, 200, 300, 400, 500) where as x-axis and z-axis positions are taken constant. The blue arrows in the Figure 5.4 shows the data forwarding direction. This simplest design with the direction of communication from sea bed to sea top, is the basis of our 2D and 3D node deployment strategies. It will automatically reduce the impact of multipath on the overall communication results.

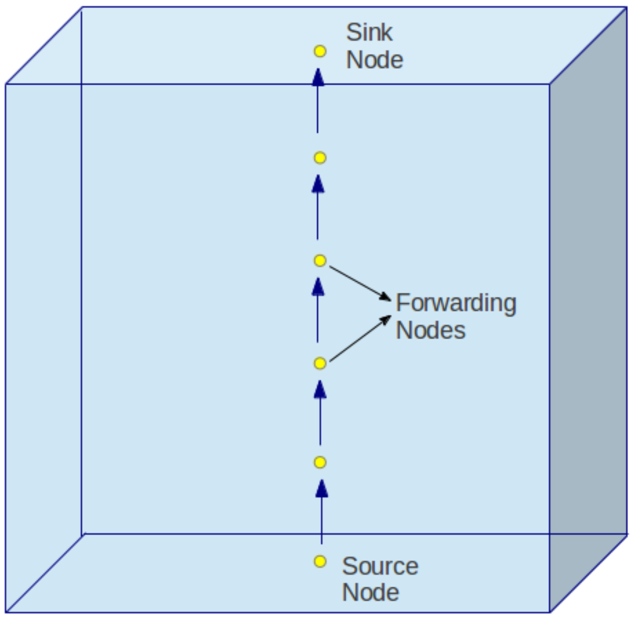


Figure 5.4: Test I (6 Sensor Nodes)

Further, in our next simulation test set-up, Test II we add 4 more nodes to Test I set-up, each in the middle of the two existing sensor nodes, so we deploy a total of 11 sensor nodes in the same underwater volume as shown in Figure 5.5. All the network parameters are same except that each node is placed at a distance of 50m from each other. The idea is to observe the network performance by decreasing the distance between the nodes and increasing the number of nodes as well as hops in a vertical line of communication.

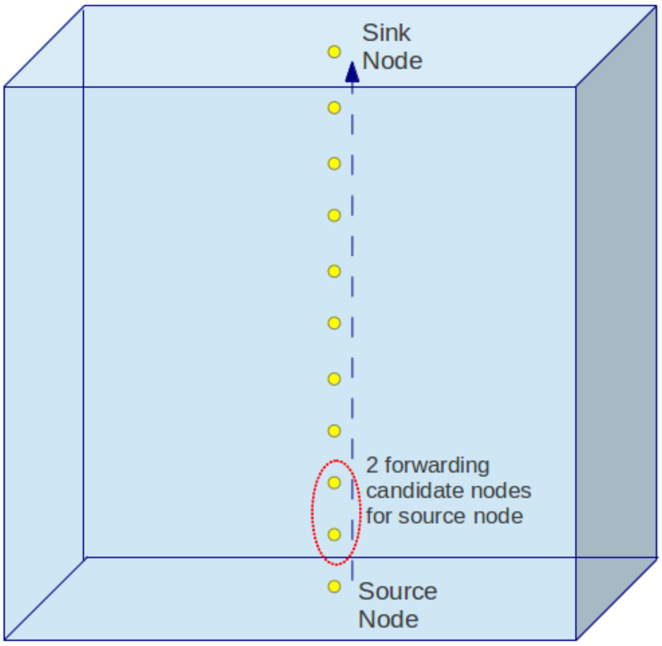


Figure 5.5: Test II (11 Sensor Nodes)

From the simulation results we generate the energy consumption plots in ns-2 as shown in Figure 5.6 and 5.7 for Test I and Test II, respectively. These plots show the actual level of energy decrement in each node for each packet transmission, hence the plots go down from initial value of energy i.e. 10000 joules. The total energy consumption of the network in Test I and Test II in transmitting 50 data packets from source node to sink node is calculated from these plots. The initial energy, 10000 joules of each node reduces with each successful transmission of data packet through it, shown by each point on the graph in Figure 5.6 and 5.7. The end points on these graphs give the left over energy in each node. By comparing the two plots in Figure 5.8, it was observed that if the distance between the sensor nodes in the network decreases, which means the number of sensor nodes increases, the total energy consumption increases. By decreasing the distance between the sensor nodes, the number of nodes involved in forwarding the same data packets through the same volume of sea is increased. Hence the overall energy consumption of the network is increased.

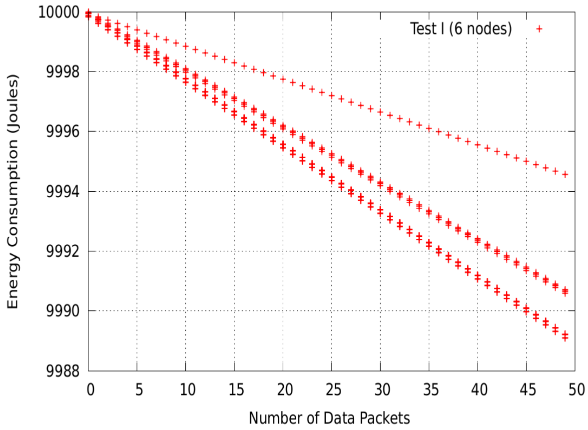


Figure 5.6: Energy Consumption Test I

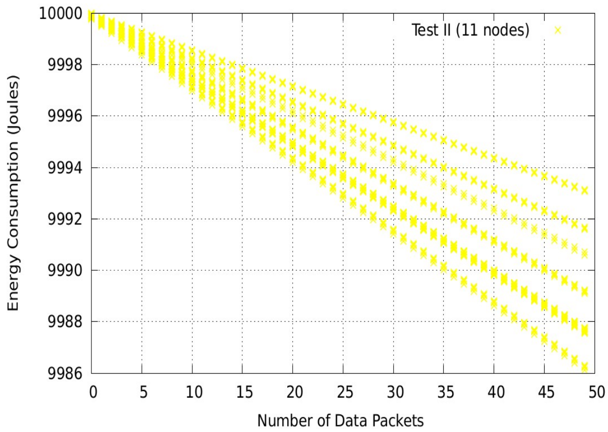


Figure 5.7: Energy Consumption Test II

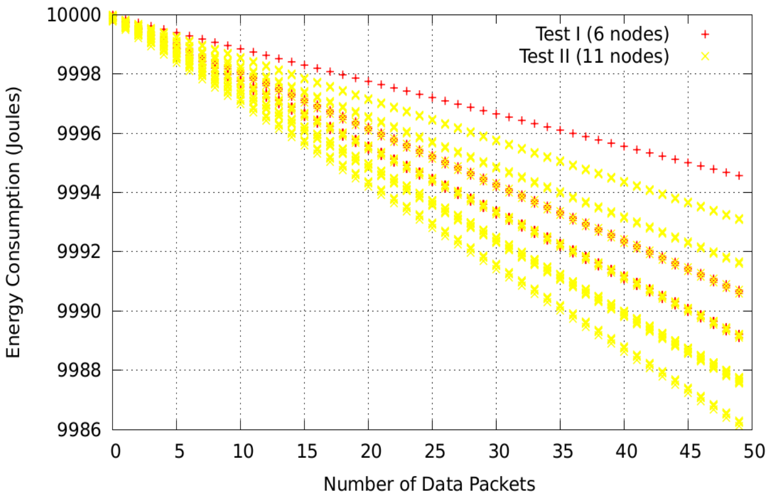


Figure 5.8: Energy Consumption Comparison - Test I & Test II

Furthermore, the simulation results show that the nodes are able to communicate successfully with each other in a straight vertical line as long as the forwarding node is within their transmission range. There is no loss of data packets as long as all the nodes are functioning properly. The plot in Figure 5.6, showing energy consumption in Test I, shows six lines if we join the points on the graph linearly. Although the lines are not very distinct, as they overlap each other. These six lines show the six nodes involved in data transmission. Each point shows the energy consumption in forwarding each data packet and the end points on the graph show the energy left in each node after the successful forwarding of 50 data packets from source node to sink node. The upper most line is the sink node and the other five lines show the energy consumption of source node and the four forwarding nodes. Sink node has least energy consumption as it is only involved in receiving the data and no transmission. Each node has initial energy of 10,000 joules and on average it reduces to 9990 joules. If we sum up all the end points of all the lines in the graph it will give us the remaining energy of the network (comprised of 6 sensor nodes) in transmitting 50 data packets from source node to sink node successfully. By subtracting the remaining energy of the network from the initial energy, we get the total energy consumption of the network in joules as shown in Table 5.1. The summary of simulation results is shown in the Table 5.1.

Table 5.1: Simulation Results Summary – Test I and Test II

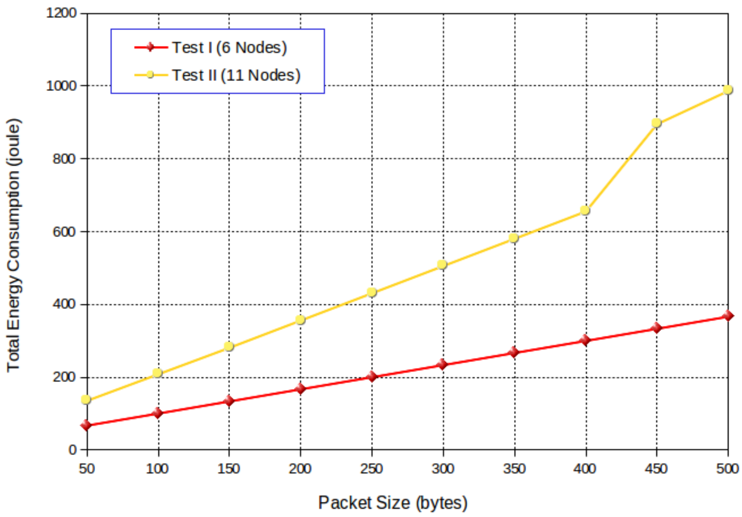
Set-up	**Test I**	**Test II**
Number of Nodes	6	11
Number of Data Packets	50	50
Total End-to-End Delay (sec)	122.7	122.7
Average Delay per Node (sec)	20.45	11.15
Average Delay per Packet (sec)	2.45	2.45
Total Energy Consumption (joule)	66.87	135.08
Average Energy per Node (joule)	11.14	12.28
Average Energy per Packet (joule)	1.33	2.70

Although total energy consumption of the network increases by increasing the number of nodes, the total end-to-end delay remains the same. It is also observed that in Test II the energy consumption per node is slightly increased. This is due to the fact that the two forwarding candidate nodes (as shown in Figure 5.5) are reachable by each node as the range of each sensor node is 100m and distance between two nodes is 50m. Therefore, when a data packet is transmitted it is received by two forwarding candidate nodes (instead of one node as in Test I) but only one of them, which is lesser in depth forwards it further. There will be duplicate reception of the data packets which increases the average energy consumption per node. Each node is able to forward the data packet to its two immediate nodes which are lesser in depth to it. Both of these forwarding candidate nodes within the range of the sending node will receive the data packet but only the one which is lesser in depth between the two will forward the data packet further. In this way if one of the forwarding candidate nodes is broken or goes missing the second one will still be able to forward the data packet and the communication path will not be broken.

We did few more tests to verify these conclusions by placing the nodes at different distances. For example 75m distance apart from each other. The number of nodes in this set-up were 8. The simulation results showed, end-to-end delay 122.7sec, same as Test I and Test II, total energy consumption 91.98 joules, average energy per node 11.49 joules and average energy per packet 1.84 joules. These results and the Test I and Test II results imply that by increasing the number of nodes in a straight line from sea bottom to sea top and decreasing the distance between them, we are increasing the total energy consumption gradually to a point where we get increased reliability. In Test II, the 50m spacing has been used to demonstrate a way of increasing reliability by doubling the number of sensor nodes in 1D. As each sensor node has a transmitting range of 100m, two nodes become available in the transmitting range of each node. This will increase reliability of the network as well as life time. The two nodes reachable by each node means each node has a spare node to continue the communication in case the main forwarding node goes missing. However if we place this node anywhere within 100m range it will only be used once the main forwarding node (which is the farthest within 100m) will go missing or broken. Hence to demonstrate reliable communication due to redundant nodes available in the network we use 50m distance between the nodes.

We analysed the energy performance of Test I and Test II for different data packet size as shown in Figure 5.9 (a) and (b). It is evident that Test II consumes more energy compared to Test I because of greater number of nodes. The energy increases linearly in both, Test I & Test II with the increase in the size of data packets. It is interesting to see that the difference in energy consumption between the two test set-ups increases gradually by increasing the size of the data packets till 400 bytes, after that the total energy consumption in Test II increases non-linearly. Which means that increasing the number of nodes can end up in huge energy overhead for larger data packet size (in this case larger than 400 bytes). This sudden increase in energy overhead is due to congestion, bottlenecks and retransmission of larger data packets in Test II. Hence in Test II scenario, for increasing the reliability the data packet size needs to be considered.

(a)



(b)

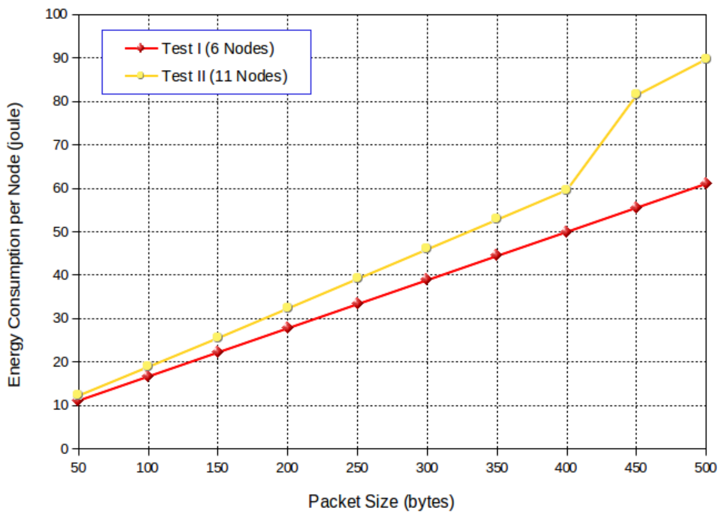


Figure 5.9: (a) Energy Consumption vs Packet Size - Test I & Test II, (b)Energy Consumption per Node vs Packet Size - Test I & Test II.

From the simulation results, we can conclude that by increasing the number of nodes in a straight line we increase the reliability at the cost of slight increase in average energy consumption per node, whereas end-to-end delay remains the same. However, increasing the packet size needs to be considered while increasing the number of nodes for reliability purpose. The vertical line of communication test set-ups (Test I & Test II) will have minimum impact of multipath as the nodes are set up for vertical communication. However if one or two of the simultaneous forwarding nodes (in Test I or Test II respectively) fail, the communication path will be broken and there will be no secondary route to transmit the data. Therefore, we extend the design to 2D and further to 3D configuration for UWSN.

5.4.2 2D Node Deployment Strategy (Test III & Test IV)

We deploy two test set-ups for 2D node deployment simulations, namely Test III & Test IV. In Test III, we deploy 12 sensor nodes as shown in Figure 5.10 in the same underwater volume i.e. $500 \times 500 \times 500 m^3$. We form two parallel paths with the forwarding nodes from source node to sink node, to increase the reliability of the network for forwarding data in 2D. In this case if one node fails in one of the paths from Source to Sink, the second path will still be forwarding the data. Hence increasing the number of nodes to double in a grid form, doubles the reliability. Here we note that the distance between the forwarding nodes is still 100m (with respect to the side, up and down node but not the diagonal node). Hence while forwarding the data each node can forward it to two neighbouring nodes, side node and up node whilst the diagonally up node will not be reachable. But only the upper neighbouring node which is lesser in depth will forward the data packet further.

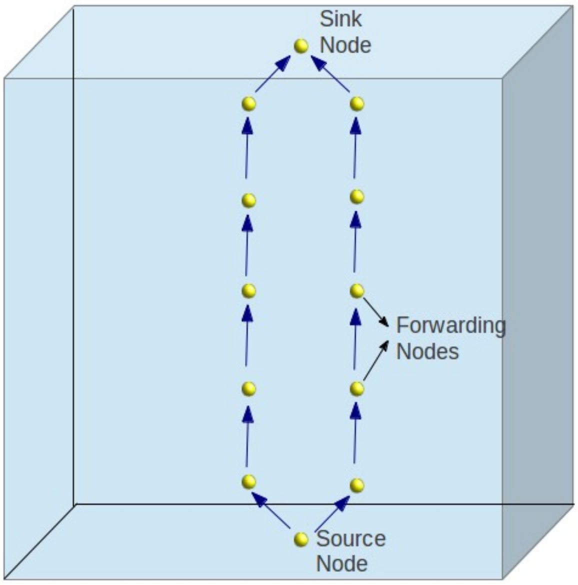


Figure 5.10: Test III (12 Sensor Nodes)

We have placed the source node and sink node in the middle of their two neighbouring nodes as shown in Figure 5.10. So that two independent data paths will be established from the source node to the sink node for communication, this set-up increases the reliability of the network. The source and sink nodes are placed on the y-axis 50m away from neighbouring nodes as shown in Figure 5.10. Their distance from the neighbouring nodes is calculated using Pythagoras theorem, which is a fundamental relation in Euclidean geometry among the three sides of a right triangle. It states that the square of the hypotenuse (the side opposite the right angle) is equal to the sum of the squares of the other two sides. Hence the distance of source and sink node from their neighbouring nodes is calculated to be 70.71m, as illustrated with a red line in Figure 5.11. The source and sink node location is set up such that this distance is less than 100m, which is the range of each node. Therefore, the forwarding nodes will be easily reachable by the source node. Similarly the sink node will be easily reachable by the forwarding nodes near the sink. Each set of four forwarding nodes make a square as the distance between each of them is set to 100m.

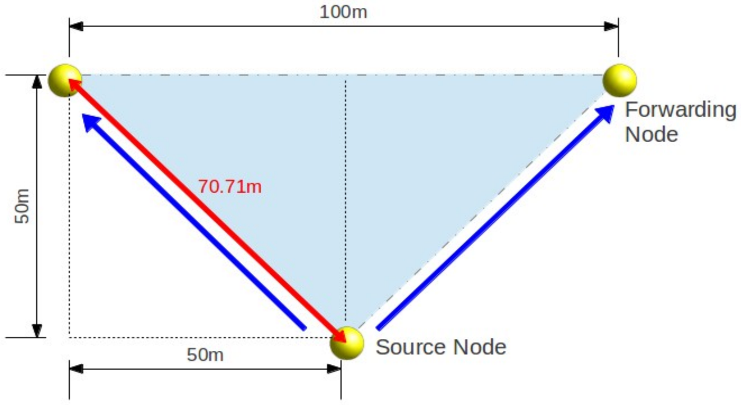


Figure 5.11: Illustration of the Distance between Source node and Forwarding Nodes

In our second 2D test set-up, Test IV, we have added four nodes to the Test III set-up. Hence Test IV set-up has a total of 16 nodes, deployed as shown in Figure 5.12. We call the additional four nodes as “Reliability Nodes”, shown in red in Figure 5.12. These nodes are similar in hardware and functionality to other nodes and their purpose is to add further reliability to the network. The network can function without these nodes as you can see in the Figure 5.10. However the Reliability Nodes provide extra stability and lifetime to the network by sharing the data traffic and providing back-up forwarding paths. Each node can now forward the data packet to three neighbouring nodes (upper, side and diagonal node). If the upper neighbouring node, which is the actual forwarding node (as being lesser in depth) fails, the diagonal node which is the

Reliability Node will continue to support the broken communication path.

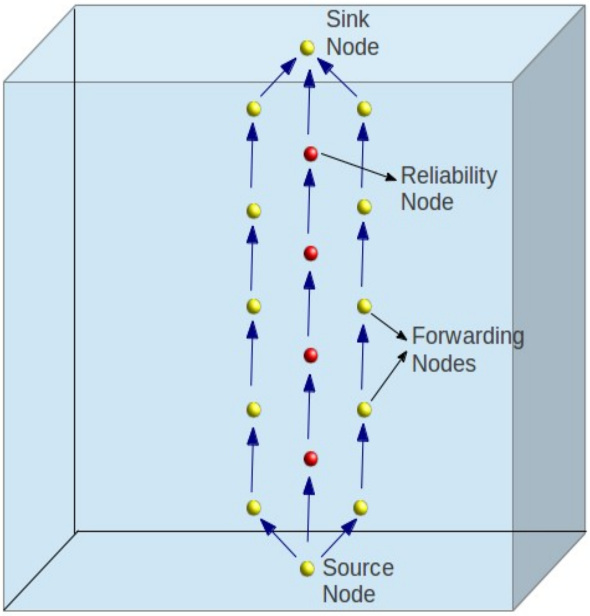


Figure 5.12: Test IV (16 Sensor Nodes)

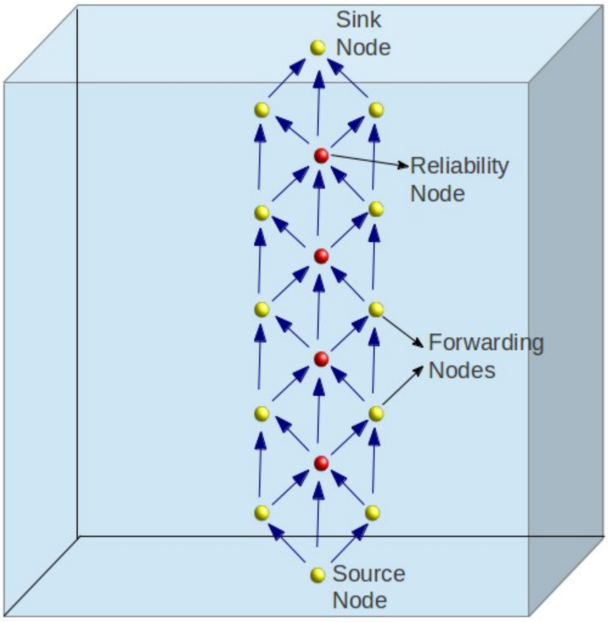


Figure 5.13: Diagonal forwarding paths in Test IV

If the communication is properly managed, the Reliability nodes can significantly increase the network lifetime by sharing the forwarding data load intelligently. This means they can dynamically collaborate to share the data load and increase the network life time. The position of these nodes in the network is such that, their role in providing the overall reliability to the network becomes highly significant. These nodes make 30% of the total number of sensor nodes in 2D test set-up. Which means increasing the number of nodes by almost 30% (from previous set-up to this) the reliability increases significantly. Each node gets another reachable node diagonally (which is lesser in depth) for forwarding the data as shown by the diagonal blue arrows in Figure 5.13.

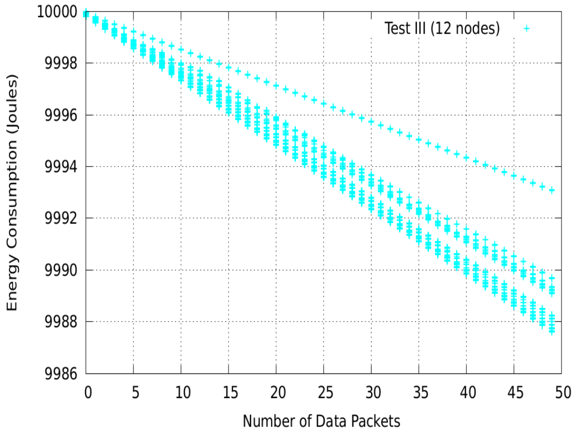


Figure 5.14: Energy Consumption Test III

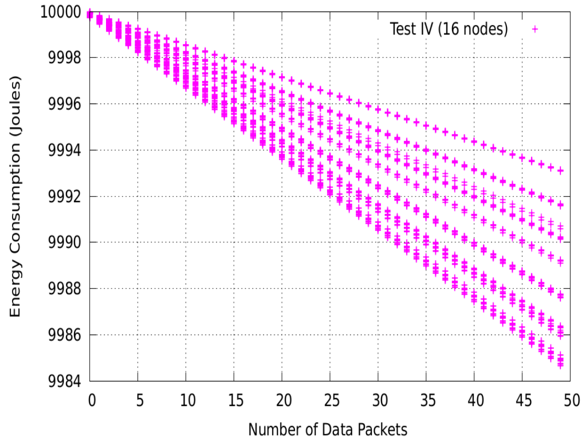


Figure 5.15: Energy Consumption Test IV

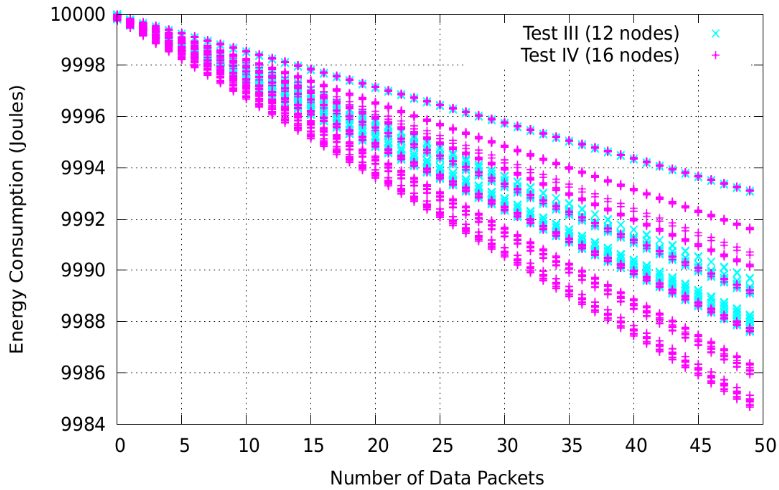


Figure 5.16: Energy Consumption Comparison - Test III & Test IV

The energy consumption plot against the number of data packets transmitted in Test III and Test IV are shown in Figures 5.14 and 5.15, respectively. Figure 5.16 shows the energy consumption comparison of the two plots. These plots show the energy consumption of each node for transmitting 50 data packets successfully. Each data packet size is 50 bytes. Each node has an initial energy of 10000 joules which is reduced when the node is in one of the three modes, transmission, reception or idle, as stated in section 5.3. The initial energy of each node reduces with each successful transmission of data packet through it, shown by each point on the graph in Figures 5.14 and 5.15. If we join the points linearly we get as many lines as the number of nodes in that network. Each line shows the energy reduction pattern of each node after successful transmission of 50 data packets. Some lines overlap as their energy reduction pattern is the same.

The simulation results for Test III and Test IV are summarized in Table 5.2. It is observed that by adding the Reliability nodes, the end-to-end delay and average energy consumption per node decreases. The total energy increases by increasing the number of nodes, but average energy per node decreases slightly due to such a node deployment strategy, which is in contrast with the results for Test I and Test II. Similarly the average delay per node and average delay per packet decreases. However the average energy consumption per packet increases as more nodes are involved in networking task.

Table 5.2 : Simulation Results Summary - Test III and IV

Set-up	**Test III**	**Test IV**
Number of Nodes	12	16
Number of Data Packets	50	50
Overall End-to-End Delay (sec)	164.26	122.7
Average Delay per Node (sec)	13.68	7.67
Average delay per Packet (sec)	3.28	2.45
Total Energy Consumed (joule)	155.13	200.4
Average Energy per Node (joule)	12.93	12.52
Average Energy per Packet (joule)	3.10	4.01

The plots in Figure 5.17 shows a comparison of simulation results of Test III and Test IV as a function of data packet size. These plots give a clear indication of how the data varies by adding the 4 additional nodes to the Test III. Figure 5.17(a) shows that the energy consumption in Test IV is higher than in Test III due to the increased number of nodes. However, the energy consumption per node is slightly less in Test IV than in Test

III as shown in Figure 5.17(b). This gives us an idea of the improvement trend in energy consumption per node by adding the Reliability Nodes and increasing the back-up paths. Basically we are providing additional data forwarding paths to the existing nodes in Test III by adding the reliability nodes in Test IV, hence increasing the reliability of the overall communication as well as the lifetime of the network at the cost of slight increase in total energy consumption. This means that there will be back-up forwarding nodes available if the main forwarding nodes fail or drain their energy. The simulation results also show that energy consumption in Test IV is linear till a data packet size of 300 bytes; after this it rises non linearly. This non linear rise in the graph indicates the congestion, bottlenecks and retransmission of data packets. Hence for adding the reliability nodes to a 2D network we need to consider the data packet size carefully not to cause congestion in the overall transmission.

(a) (b)

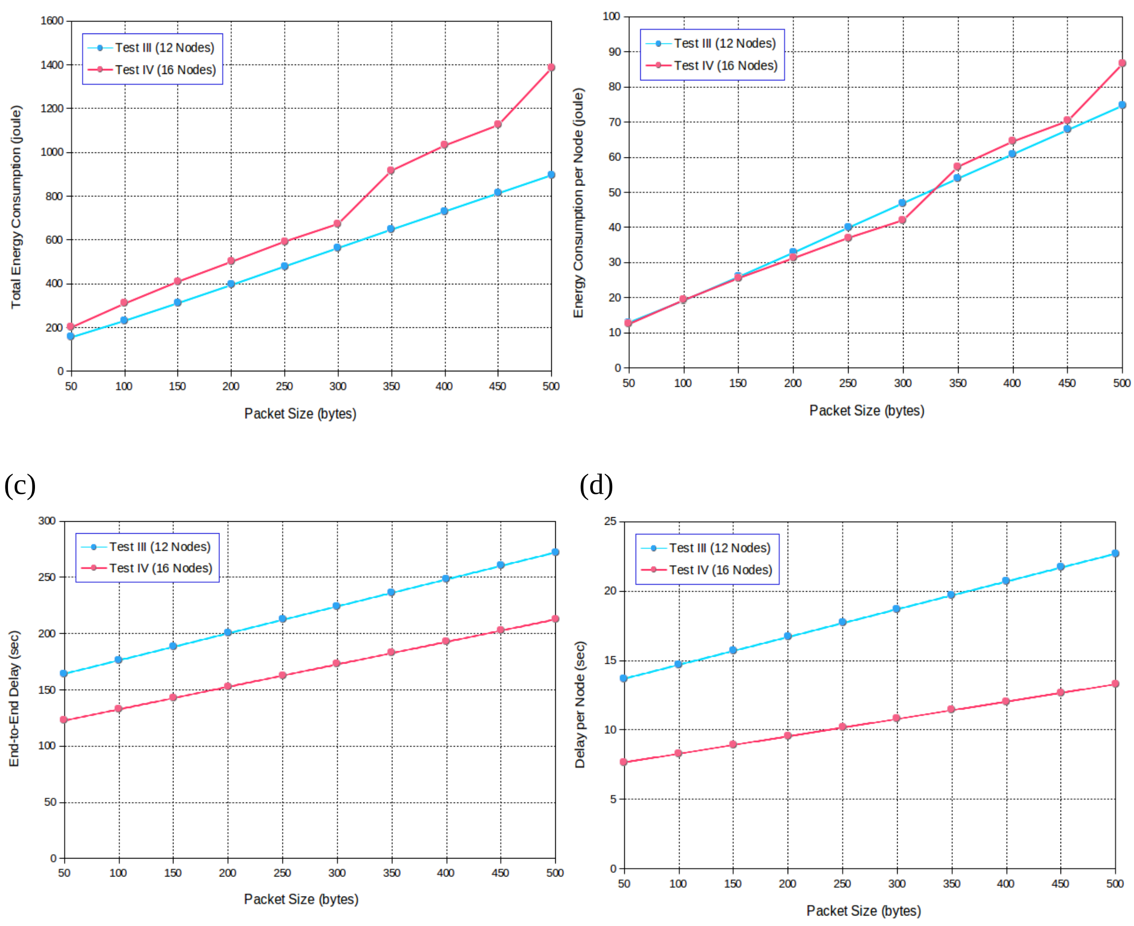


Figure 5.17: Simulation Results Comparison - Test III & Test IV.
(a) Total Energy Consumption (b) Energy Consumption per Node,
(c) Total End-to-End Delay, (d) Delay per node

It is interesting to see in Figure 5.17(c) and (d) that by adding the Reliability nodes to Test III we see a significant improvement in end-to-end delay in Test IV. The end-to-end delay plot of both Test-III and Test IV is linear and the difference between the two increases by increasing the size of data packets. This increase of the difference is more obvious in the end-to-end delay per node plot and increases linearly with the increase in data packet size as shown in Figure 5.17 (d).

5.4.3 3D Node Deployment Strategy (Test V & Test VI)

In the next test set-up, we propose a 3D node deployment strategy based on the 2D configuration, which is simple yet effective and will improve the overall 3D network performance. In 3D test set-up, Test V and Test VI, we place the sensor nodes in the form of a 3D cubic lattice structure to cover the desired underwater volume. This is done by stacking up cubes to cover the desired underwater volume, with sensor nodes placed at the vertices of each cube.

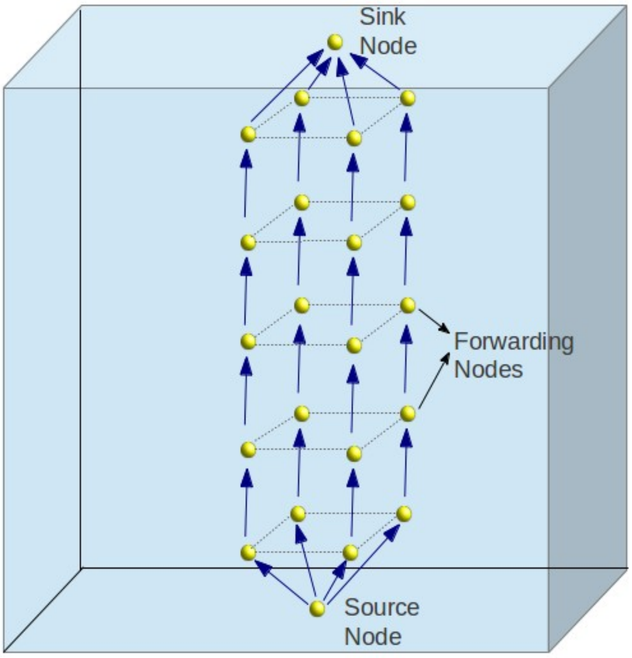


Figure 5.18: Test V (22 nodes)

Figure 5.18 shows a unit structure from such a network. The source node and sink node are 50m below and above respectively from the middle of the plane of their nearest cube. The source node is placed 50m below the middle of the lower plane of the cube nearest to the source node (which is the lowest cube in the structure) as shown in Figure 5.18 and the sink node is placed 50m above the middle of the upper plane of the cube nearest to the sink node (which is the top most cube in the structure) as shown in Figure 5.18. Test V set-up is comprised of 22 sensor nodes.

For the Test VI set-up, a Reliability node is placed in the middle of each cube of Test V, as shown in Figure 5.19, hence a total of 26 nodes. In this configuration each cube is formed by six pyramids with their bases making six planes of the cube and peaks meeting at the centre of the cube as shown in Figure 5.20. The sensor nodes are placed at the vertices of these pyramids or at the vertices of the cube, and one in the middle of the cube, which is the point where the pyramids top vertex coincide as shown in Figure 5.21. It is understood that if a pyramid's vertex coincides with the vertex of another pyramid (which is the case here) then only one sensor node is placed there. Similarly if the vertex of a cube coincides with another cube (which is the case here as cubes are stacked up to form a 3D cubic lattice in the desired area) then only one sensor node is placed there.

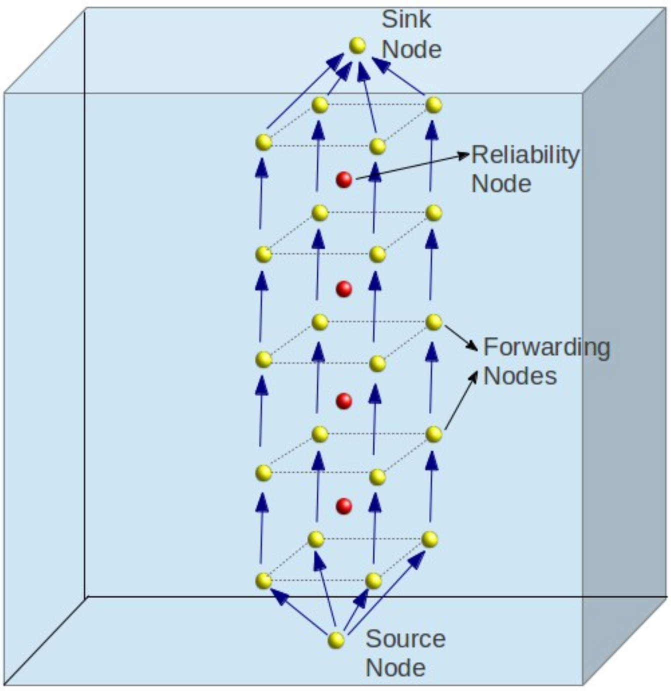


Figure 5.19: Test VI (26 Nodes)

The distance of the Reliability node with respect to its four immediate forwarding nodes, which are the nodes placed at the vertices of the plane above the Reliability node, is 86.6m. This has been calculated using the formula for calculating the distance between two points in 3D as below,

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad ,$$

where (x_1, y_1, z_1) are the 3D coordinates of one of the diagonal nodes and (x_2, y_2, z_2) are the 3D coordinates of one of its forwarding nodes. As the range of each node is 100m, the immediate neighbouring nodes of each diagonal node are well within its range (as the calculated distance between them is 86.6m).

In this way not only source and sink nodes are accessible to four immediate nodes, each sensor node in this set-up is also reachable to four forwarding nodes. Two at the sides, one at the top and one at the diagonal (which is the Reliability node). Figure 5.19 can be assumed as a cell (unit structure) taken from a network comprising of such cubic lattice to cover the desired monitoring volume of the sea. It shows that Reliability nodes make approx. 15% of the total number of nodes. However, due to their positioning they play a significant role in increasing the reliability by increasing the forwarding paths for communication within the network. This means in this configuration by adding 15% more nodes to the network we provide several back-up communication paths which can take up a vital role if one or more nodes go missing, moves out of range or fail and main forwarding link is broken. The Reliability nodes do not require stronger batteries compared to other nodes.

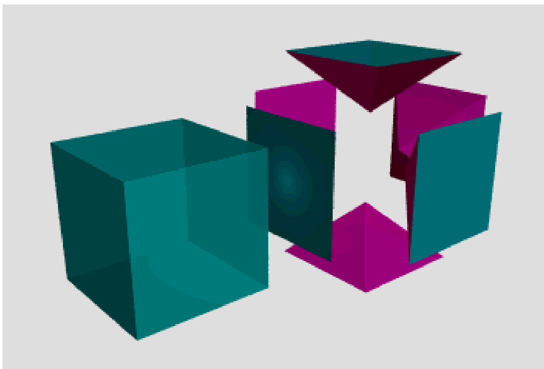


Figure 5.20: Six Pyramids Forming a Cube

In the 3D cubic lattice design, several structures like the one shown in Figure

5.19 will be stacked together to occupy the desired underwater volume. Considering the entire network comprising of such unit structure cells, it is observed that these nodes are more likely to be reached by the four nodes in the lower plane if the main forwarding node of one of these four nodes is broken or goes out of reach. Here it is noted that each node at the vertex of the cube will be reachable to the upper four of these diagonal Reliability nodes (for data forwarding) each in the middle of the four cubes adjacent to that node in the 3D cubic lattice, a slice of such lattice is shown in Figure 5.22. Each node in the plane of the cube can access four Reliability nodes diagonally in the upper four adjacent cubes, to that node.

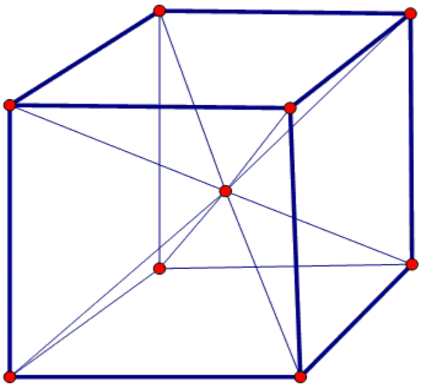


Figure 5.21: Node Deployment in the Cubic Lattice Unit Cell

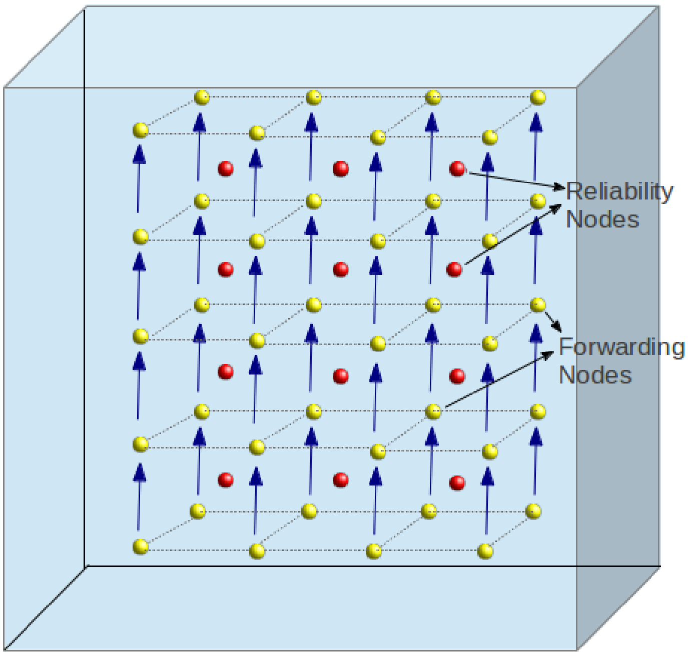


Figure 5.22: Slice of the 3D Node Forwarding Node Deployment

The energy consumption plots generated from the simulation results for Test V and Test VI are shown in Figure 5.23 and Figure 5.24 respectively. These plots show the energy consumption of each node for transmitting 50 data packets successfully. Each data packet size is 50 bytes. Each node has an initial energy of 10000 joules which decreases when the node is in one of the three modes, transmission, reception or idle, as stated in section 5.3.

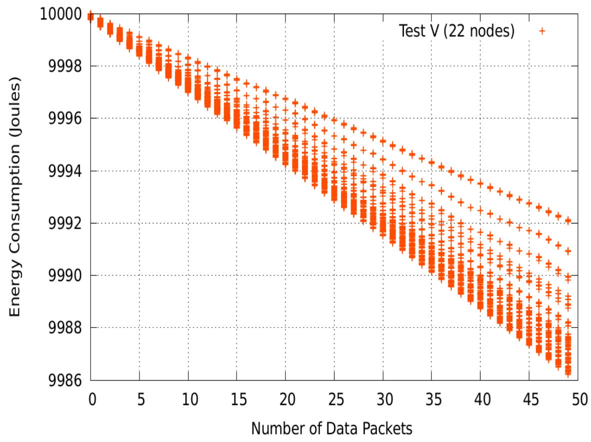


Figure 5.23: Energy Consumption Test V

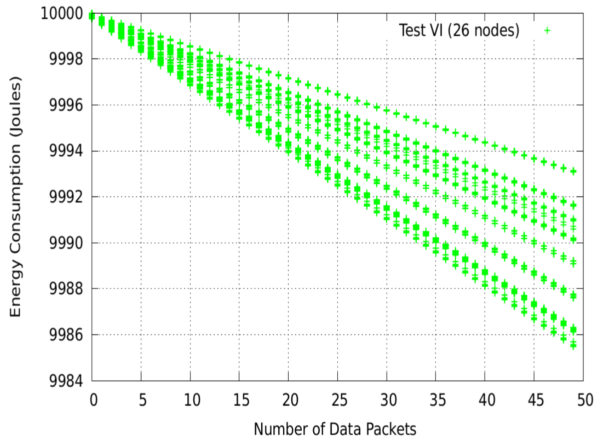


Figure 5.24: Energy Consumption Test VI

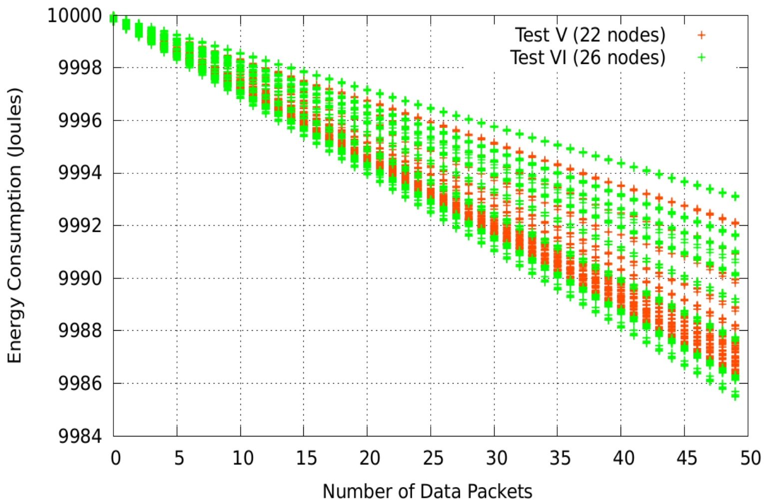


Figure 5.25: Energy Consumption Comparison - Test V & Test VI

The initial energy of each node reduces with each successful transmission of data packet through it, shown by each point on the graph in Figure 5.23 and 5.24. If we join the points linearly, we get as many lines as the number of nodes in that network. Each line shows the energy reduction pattern of each node after successful transmission of 50 data packets. Some lines overlap as their energy reduction pattern is the same. Figure 5.25 shows a comparison of the two plots of Figure 5.23 and Figure 5.24. These plots show that by adding the Reliability Nodes in the 3D node deployment strategy the total energy consumption remains almost the same instead of increasing due to increased number of nodes. This means that we are adding significant reliability to the network at no extra energy cost. And the energy consumption per node has decreased as the number of nodes is increased. This indicates that by 3D node deployment strategy in Test VI we are distributing the energy cost over the network in order to increase the life time of the network.

The simulation results for Test V and Test VI are summarized in Table 5.3. It is observed that by adding the Reliability nodes to the 3D node deployment strategy, the end-to-end delay and average energy consumption per node decrease. The total energy consumption decreases slightly even though the number of nodes is increased. The average energy consumption per node decreases to even less than the average energy per node in Test III and IV. This average energy per node in Test VI is comparable to the average energy per node in Test I. Hence in this effective 3D node deployment strategy the energy consumption per node is as less as in the simplest test set-up, Test I. The total end-to-end delay decreases in Test VI as compared to Test V. Hence the average delay per node and the average delay per packet decreases significantly.

Table 5.3 : Simulation Results Summary - Test V and VI

Set-up	**Test V**	**Test VI**	**Test VI-** 75m	**Test VI-** 50m
Number of Nodes	22	26	26	26
Number of Data Packets	50	50	50	50
Overall End-to-End Delay (sec)	148.34	122.7	114.3	70.5
Average Delay per Node (sec)	6.74	4.71	4.39	2.71
Average delay per Packet (sec)	2.97	2.45	2.29	1.41
Total Energy Consumed (joule)	305.66	298.85	253.34	219.12
Average Energy per Node (joule)	13.89	11.49	9.7	8.43
Average Energy per Packet (joule)	6.1	5.98	5.07	4.38

We performed few more tests in which we changed the distance between the sensor nodes in Test VI from 100m to 75m and then 50m. The transmission range of each node is still the same (100m). The results of these tests are summarized in Table 5.3 (as Test VI-75m and Test VI-50m respectively). The number of nodes and data packets are same as in Test VI, only the distance between the nodes is reduced. The simulation results of Test VI-75m showed, end-to-end delay 114.3sec, which is slightly decreased by reducing the distance between the nodes from 100m (as in Test VI) to 75m (as in Test VI – 75), this is due to the reason that the overall distance covered by the packets is reduced compared to when they were placed 100m apart. Hence the packets will reach the destination node quicker due to less overall distance. The total energy consumption is also reduced to 253.34 joules form 298.85 joules by reducing the distance from 100m to 75m. In the simulation results of Test VI-50m, the end-to-end delay is reduced to 70.5 sec, this is due to the reason that two nodes are available within the reach of each node (with 100m transmission range) and only the farthest node which is least in depth from source node towards destination node will be chosen as best forwarding node, therefore reducing the end-to-end delay. However the total distance from source to destination node is also halved due to 50m spacing between the nodes compared to Test VI, in which nodes are 100m apart from each other. The total energy consumption also shows this impact as it is reduced to 219.12 joules. The overall energy consumption is not reduced to half by reducing the distance between the nodes from 100m to 50m because the transmission range of each node is still 100m, only reception of each packet is reduced by 50m. There will also be impact of congestion and duplicate transmissions towards the reachable nodes.

These results show that by decreasing the distance between the nodes (lesser than their transmission range) the end-to-end delay decreases as well as the overall distance travelled by the packets from source node to destination node. Similarly the total energy consumption is also reduced as the packets are not reaching as far as they can with the transmission range of each node. Hence if we place the nodes closer to each other they will still communicate but we will not be utilizing the maximum capacity of the network.

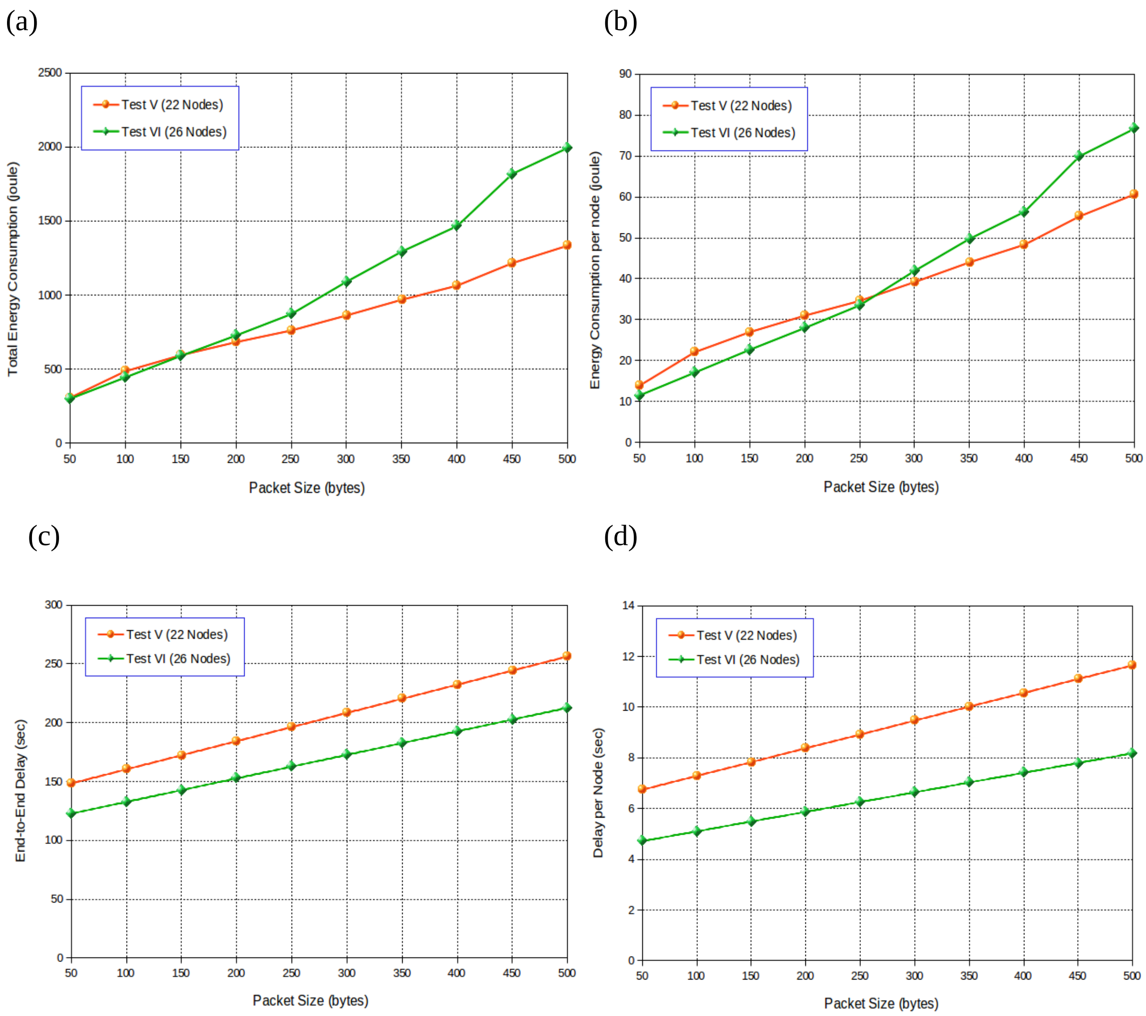


Figure 5.26: Simulation Results Comparison - Test V & Test VI.
(a) Total Energy Consumption (b) Energy Consumption per Node,
(c) Total End-to-End Delay, (d) Delay per node

The plots in Figure 5.26 show the comparison of the simulation results of Test V and Test VI as a function of data packet size. These plots give a clear picture of how the data varies in 3D node deployment strategy by adding the 4 additional nodes in Test VI. Figure 5.26(a) shows that the total energy consumption remains the same for Test V and VI up to a data packet size of almost 160 bytes. After that we can see the increase in total energy consumption which indicates the impact of increased number of nodes in Test VI. The difference in energy consumption between Test V and Test VI increases almost linearly till the packet size of 400 bytes and after that the energy consumption in Test VI becomes non linear. This indicates congestion, bottlenecks and retransmission. Figure 5.26(b) shows the energy consumption per node in Test V and Test VI, it is lesser in Test VI than Test V up to the packet size of 250 bytes, then it coincides at packet size of almost 260 bytes and increases linearly up to 400 bytes. After 400 bytes data packet size the energy consumption per node in Test VI becomes non linear. This indicates that by adding the reliability nodes we get better energy performance up to 250 bytes data packet size. Basically we are providing additional data forwarding paths to the existing nodes in Test V by adding the reliability nodes in Test VI, hence increasing the reliability of the overall communication as well as the lifetime of the network by reducing the energy cost per node for reasonably large data packet size. This means that there will be back-up forwarding nodes available if the main forwarding nodes fail or drain their energy. However by adding the reliability nodes to a 3D network we need to consider a reasonable data packet size so as to avoid congestion in the overall transmission.

It is interesting to see in Figure 5.26(c) and (d) that by adding the Reliability nodes to Test V, a significant improvement in end-to-end delay in Test VI is achieved. The end-to-end delay plot of both Test V and Test VI is linear and the difference between the two remains constant by increasing the size of data packets. However the difference between the end-to-end delay per node in Test V and Test VI increases linearly with the increase in data packet size.

Further we examine the comparison of simulation results of Test VI against Test VI-75 and Test VI-50 in Figure 5.27, in order to give further insight into the analysis of data packet size with respect to distance between the nodes. These plots show how total energy and end-to-end delay varies in our 3D node deployment strategy for different distances between the nodes or we can say for different size of the cube in cubic node deployment, keeping the transmission range and number of nodes constant. Figure 5.27(a) & (b) clearly indicates that by reducing the size of the cube in 3D cubic deployment strategy, the energy and end-to-end delay also decreases however it is understood that the overall distance a packet reaches is also significantly reduced. By increasing the packet size more than 200 bytes, the total energy consumption plots become non-linear. Each rise in the plots in Figure 5.27(a), shows increase in energy consumption in order to handle congestion caused by larger data packet size and each drop in these plots indicate packet loss. In Test VI-75 for data packet size greater than 200 bytes, there is congestion till 400 bytes data packet size after which there is drop in energy consumption due to packet loss. In Test VI-50 there is less congestion and packet loss as there are two reachable nodes for each forwarding node, however the total distance covered by each packet is half of the distance covered in Test VI, compromising the overall efficiency of the network.

(a) (b)

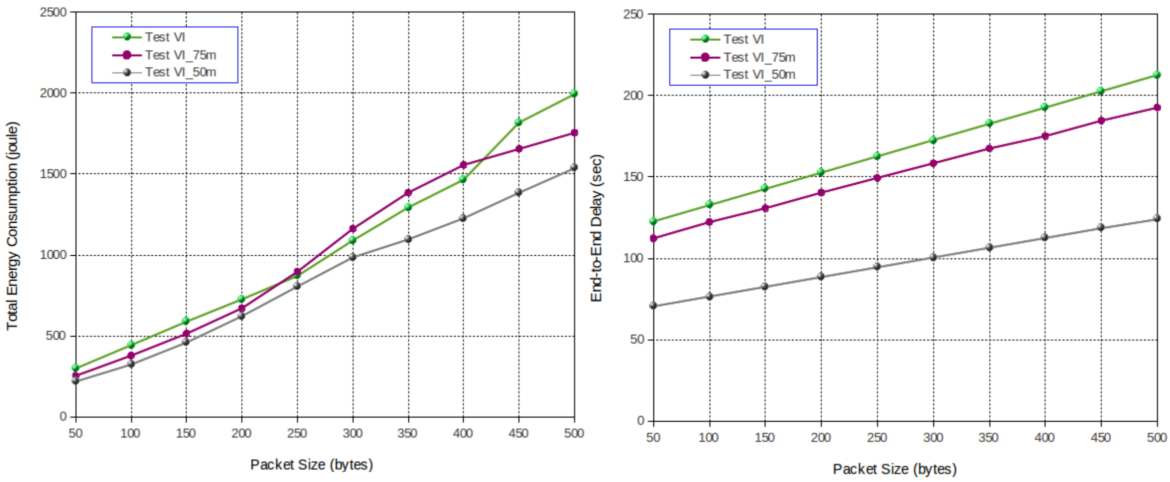


Figure 5.27: Simulation Results Comparison - Test VI, Test VI_75 & Test VI_50.
(a) Total Energy Consumption (b) Total End-to-End Delay

Cube deployment is the simplest 3D deployment strategy used for the illustration of this concept, other shapes like hexagonal prism deployment, rhombic dodecahedron deployment and truncated octahedron deployment etc. can benefit from this deployment strategy. In general any 3D node deployment configuration can benefit from the significant reliability added by intelligent placement of Reliability Nodes in the middle of each cell of the 3D lattice structure.

5.4.4 3D Random Node Deployment (Test Rand)

It is a common practice in the research community to analyse any networking protocol by random node deployment methods. Although random node deployment methods are common and best suited for some of the terrestrial WSN applications, they are not the best option for UWSN. In UWSN, sensor nodes are not only bigger in size but are expensive. Deploying them randomly will have extremely low probability of best resource utilization. To verify this we compare our 3D node deployment strategy performance with the random 3D node deployment to determine how it would affect the overall network performance in terms of energy and end-to-end delay.

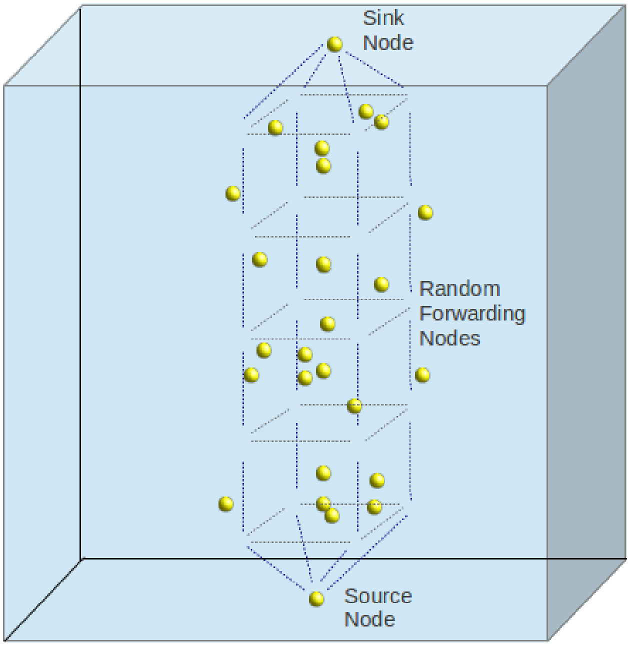


Figure 5.28: Test Rand (26 Nodes)

We have deployed the same number of sensor nodes as in Test VI but randomly as shown in Figure 5.28 and name this test set-up as Test Rand. The boundary conditions of the random deployment is to ensure a connectivity between the source node and the sink node. The energy consumption plot generated from the simulation results of the Test Rand is shown in Figure 5.29. The plot shows the energy consumption of each node for transmitting 50 data packets successfully. The initial energy, 10000 joules of each node decreases with each successful transmission of data packet through it, shown by each point on the graph in Figure 5.29. Each data packet size is 50 bytes. Joining the points linearly in the plot, we get 26 lines representing the energy consumption pattern of the 26 sensor nodes involved in the network communication. Some lines overlap as their energy reduction pattern is the same.

Figure 5.30 shows a comparison of the energy consumption plot of Test VI, which is our proposed 3D node deployment strategy and Test Rand. These plots show that the energy consumption is increased in the random deployment. Although reliability of the network transmission has been adversely affected due to random deployment as there might only be one eligible forwarding node for certain hops and when the battery of such a node is drained or this node fails or goes out of range there will be no other back up node and the communication link will be broken. This indicates that by 3D random node deployment strategy we not only increase the overall energy consumption but we seriously compromise on reliability of the network for forwarding the data packets.

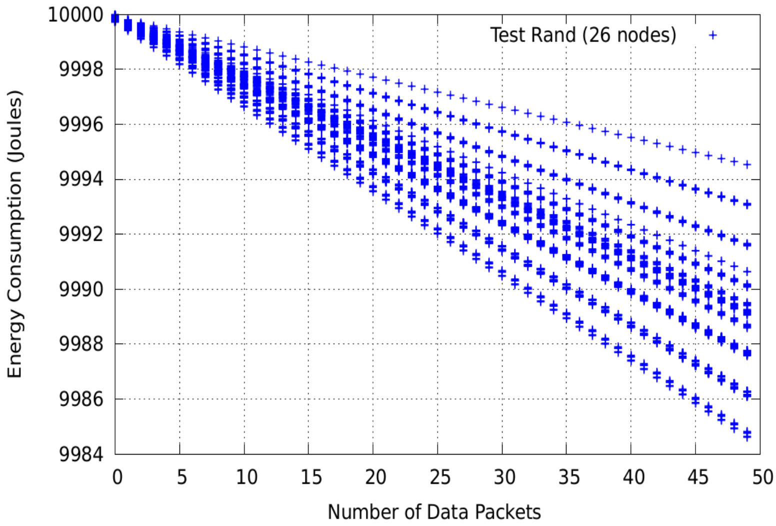


Figure 5.29: Energy Consumption Test Rand

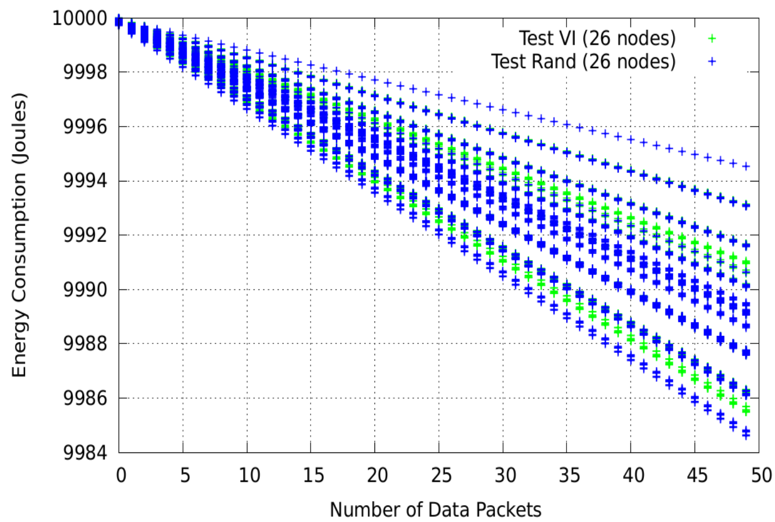


Figure 5.30: Energy Consumption Comparison - Test VI & Test Rand

To verify these results, we performed several random deployment tests, in order to find their mean and standard deviation values. Figure 5.31 shows the normal distribution and standard normal distribution of the random deployment tests for the total energy consumption and end-to-end delay values.

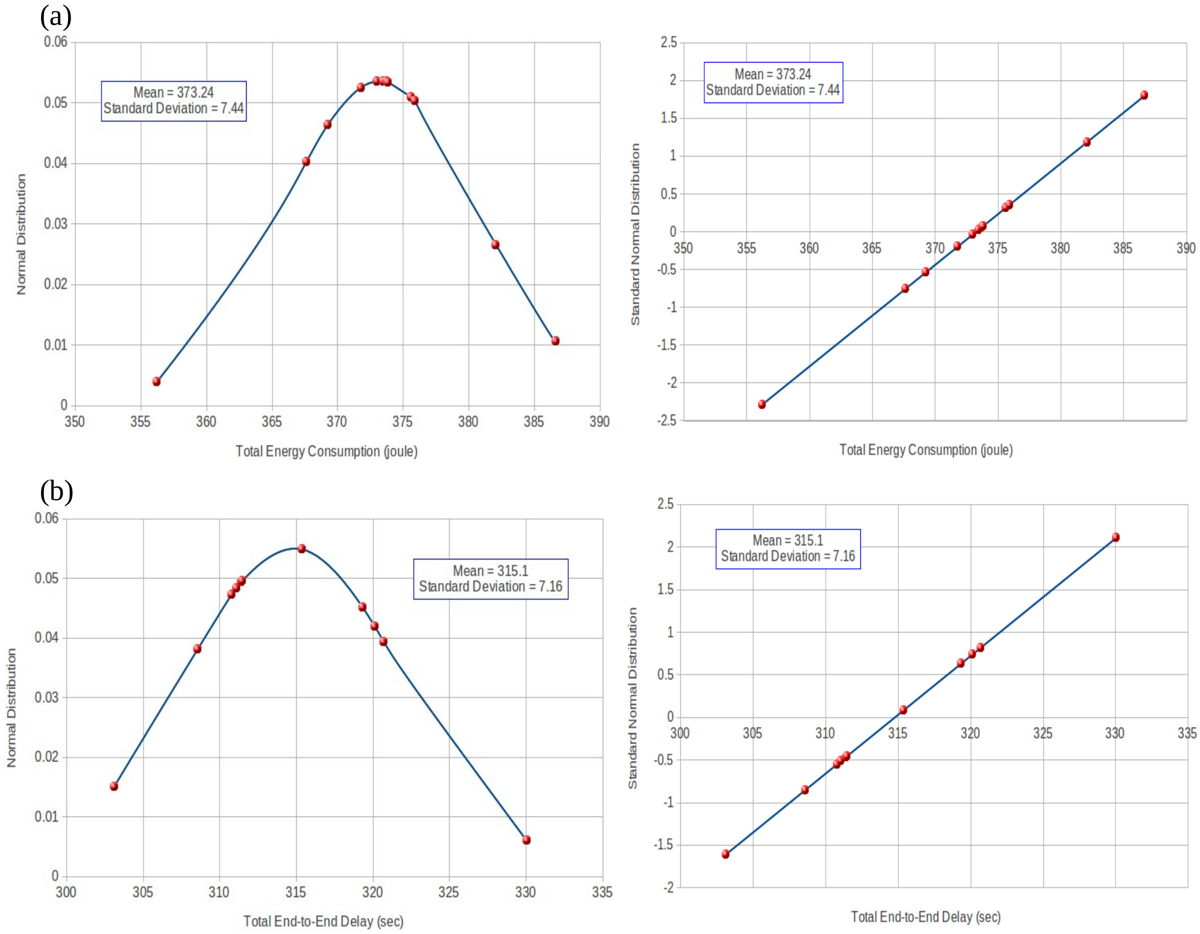


Figure 5.31: Normal Distribution and Standard Normal Distribution – Test Rand.
(a) Total Energy Consumption (b) Total End-to-End Delay

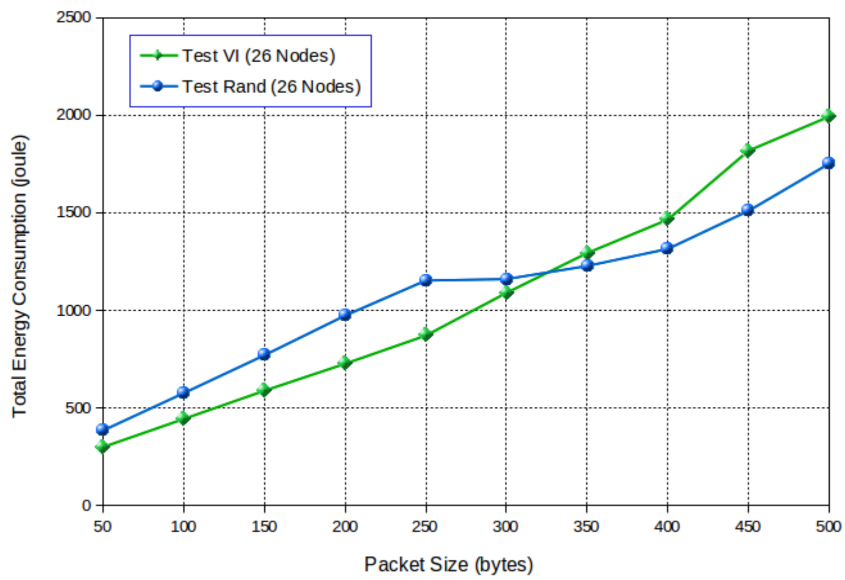
The simulation results summary for the mean values of Test Rand against Test VI is given in Table 5.4. For Test Rand_mean, we have taken the mean values for end-to-end delay and energy consumption generated through several random deployment tests. It is observed that if we deploy the nodes randomly in 3D set-up the total energy consumption as well as average energy consumption increases by 25%. Furthermore the total end-to-end delay and average delay increases drastically by more than 150%. This is a high impact increase and can result in the poor performance of the networking protocols in 3D, no matter how efficiently they are designed to work. Normally networking protocols in UWSN are tested in 2D random deployment. However the actual underwater environment is 3D and it is not even imagined how drastically the results can change in 3D environment with random deployment.

Table 5.4 : Simulation Results Summary - Test Rand_mean against Test VI

Set-up	**Test VI**	**Test Rand_mean**
Number of Nodes	26	26
Number of Data Packets	50	50
Overall End-to-End Delay (sec)	122.7	315.1
Average Delay per Node (sec)	4.71	12.11
Average delay per Packet (sec)	2.45	6.30
Total Energy Consumed (joule)	298.85	373.24
Average Energy per Node (joule)	11.49	14.35
Average Energy per Packet (joule)	5.98	7.46

The plots of Figure 5.32 show a comparison of the simulation results of Test VI and the first random deployment test, Test Rand as a function of data packet size. These plots give a clear picture of how the data varies in 3D environment for our proposed 3D node deployment strategy and the random node deployment strategy. Figure 5.32(a) shows that the total energy consumption for Test Rand is almost 25% more than Test VI initially and increases upto almost 32% till the data packet size of 250 bytes. The difference in energy consumption between Test Rand and Test VI increases up to the packet size of 250 bytes and after that the energy consumption in Test Rand drops and becomes non linear. This indicates packet loss due to congestion and no alternative back-up paths. It means that the random node deployment, under the boundary conditions to provide connectivity from the source node to the sink node may provide connectivity but will have a serious impact on the reliability by having no alternative paths for some of the sensor nodes. Whereas for the total energy consumption in Test VI we get a linear plot up to a data packet size of 400 bytes and after that the energy plot increases non linearly but there is no depreciation in the graph. This shows that for larger data traffic there will be congestion, bottlenecks and retransmissions but lower packet loss as there are sufficient alternative paths available for the transmission of data in Test VI. Therefore the 3D node deployment strategy in Test VI ensures reliable communication even with the increase of data packet size. Basically we are providing additional data forwarding paths by adding the reliability nodes at no extra cost to the network resources compared to a normal 3D cubic grid deployment as in Test V. Our 3D node deployment strategy not only increases the reliability of the overall communication it also increases the lifetime of the network by reducing the energy cost per node for reasonably large data packet size. This means that there will be back-up forwarding nodes available if the main forwarding nodes fail or drain their energy. However for data traffic size larger than 350 bytes we need to consider the energy cost for its transmission which will affect the overall network lifetime.

(a)



(b)

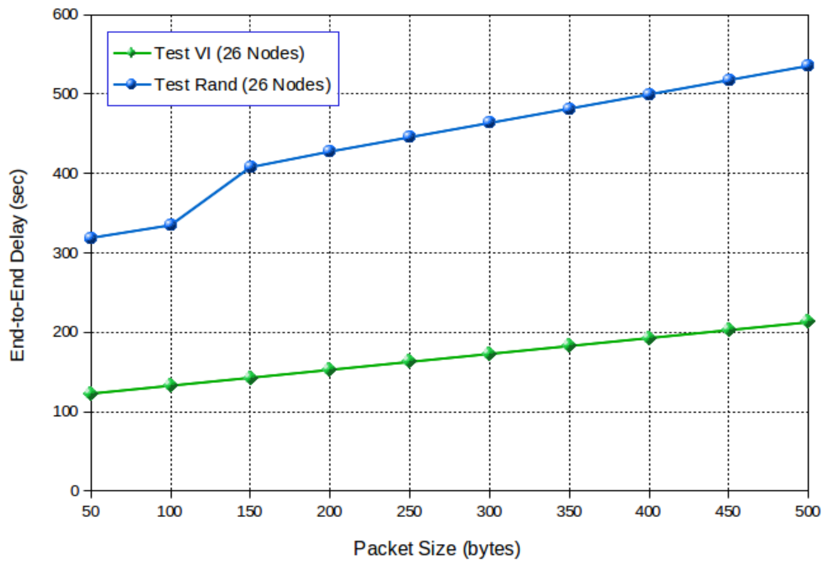


Figure 5.32: Simulation Results Comparison - Test VI & Test Rand.
(a) Total Energy Consumption (b) Total End-to-End Delay

It is seen in Figure 5.32(b) that random node deployment in 3D adversely affects the end-to-end delay. With an increase of more than 150% for the data packet size up to 100 bytes and after that it increases even further. Where as our 3D node deployment strategy ensures least end-to-end delay which is the same as for the Test I (least end-to-end delay possible in this 3D scenario).

5.5 Conclusion

In this chapter we have proposed a 3D node deployment strategy for UWSN in order to improve the networking protocol performance. We have developed it step by step from a single line of communication to an effective 3D node deployment. We observed the impact of Reliability nodes, a term introduced in our design for nodes providing reliability and increasing the lifetime of a network. We analysed and tested our strategy through several simulations. And verified that our design provides reliability to the UWSN at no extra cost to the resources. Finally, we compared our results to random node deployment in 3D, which is commonly used for analysing the performance of UWSN protocols. The comparison results verified our effective deployment design and showed that it provides almost 25% less energy consumption to the random deployment and almost 150% less end-to-end delay. It also revealed that by increasing the data traffic, our 3D node deployment strategy has least loss of data due to several back-up paths available which is in contrast to random deployment where the packet loss occurs by increasing the data traffic.

This thesis work introduces the novel concept of reliable 3D sensor node deployment approach instead of random node deployment schemes adapted from terrestrial WSN. This approach improves the networking protocol performance significantly as revealed by the simulation results in this research work. In 3D underwater environment it is extremely difficult to ensure full sensing as well as communication coverage and reliable transmission by random node deployment strategies. However, this has been achieved by our novel concept of carefully deploying the sensor nodes in a 3D node deployment strategy, providing several back-up paths to combat dynamic network configuration and node mobility in an UWSN. This strategy can be embedded with any routing protocol design and can offer significant improvement in its performance.

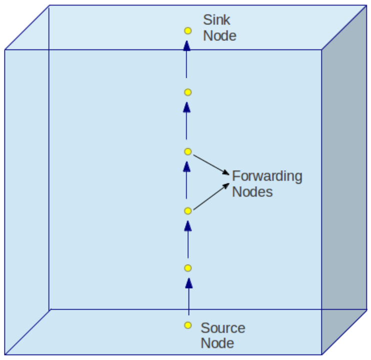
CHAPTER 6

Performance Analysis of 3D Node Deployment Strategy

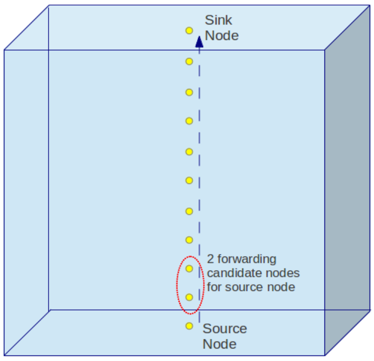
6.1 Introduction

This chapter revolves around the simulation experimental results to analyse the performance of the 3D node deployment strategy, Test VI, presented in Chapter 5, against all other test-set ups, presented in Chapter 5, namely Test I, Test II, Test III, Test IV, Test V and Test Rand. Figure 6.1 shows the summary of all the test set-ups presented in Chapter 5. Several simulation experiments revealed some interesting facts to relate among the seven test set-up scenarios. In Chapter 5, the test set-ups within each of the deployment design strategies (i.e. vertical line of communication, 2D deployment and 3D deployment design strategy) were compared and analysed. In this chapter, all the seven test set-ups arising from all the deployment design strategies presented in chapter 5 will be compared and analysed all together. The total energy consumption, energy consumption per node, total end-to-end delay and delay per node for all the test set-ups is analysed and discussed in this chapter.

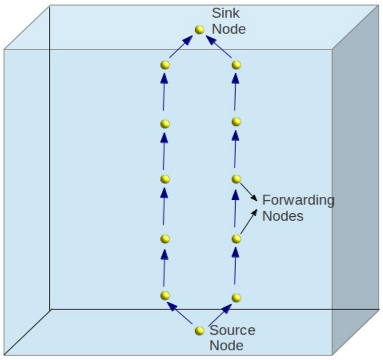
(i)



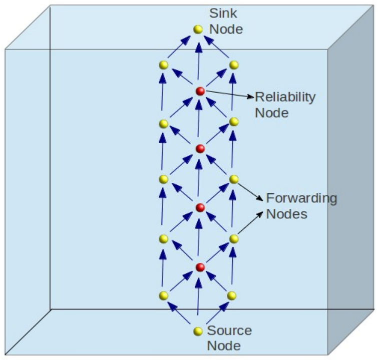
(ii)



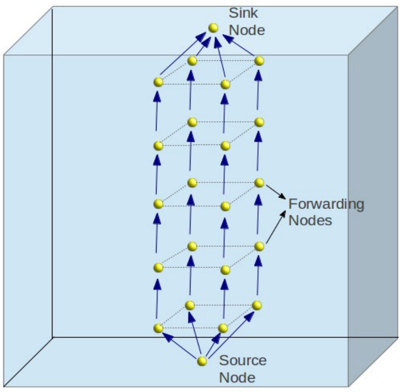
(iii)



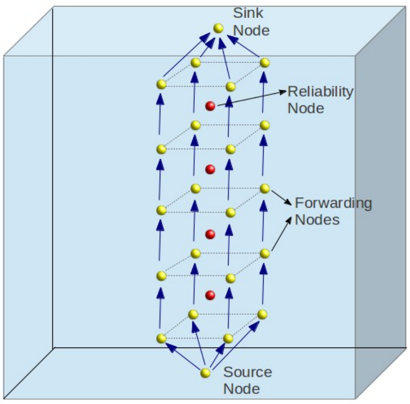
(iv)



(v)



(vi)



(vii)

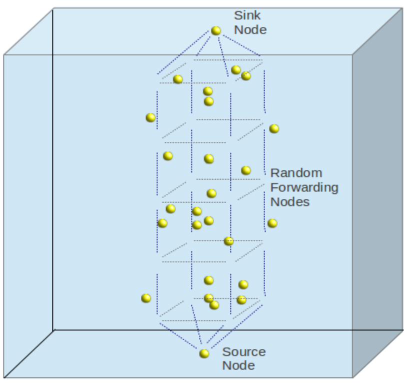


Figure 6.1: Experimental Test set-ups:

(i) Test I – 6 Nodes – 1D
(ii) Test II – 11 Nodes – 1D
(iii) Test III – 12 Nodes – 2D
(iv) Test IV – 16 Nodes – 2D
(v) Test V – 22 Nodes – 3D
(vi) Test VI – 26 Nodes – 3D
(vii) Test Rand – 26 Nodes – 3D

From several simulation results, it is observed that the 3D node deployment approach, presented in this research work, which is Test VI stands out and Test Rand, which is the commonly used random 3D node deployment approach for testing the performance of any networking protocol has significant drawbacks. These drawbacks can affect the overall network performance adversely. No matter how efficient the networking protocols are designed to perform, testing them by random node deployment approach cannot judge their efficiency properly. On the other hand, Test VI is simple to implement and performs much better than other test set-ups in terms of energy consumption and delay and ensures better connectivity, reliability and coverage. It even performs better than the 2D test set-ups (Test I to Test IV) when compared on per node basis.

6.2 Total Energy Consumption

The total energy consumption comparison of all the test set-ups for different data packet size is shown in Figure 6.2. Since all the test set-ups have different number of nodes, it reveals some interesting facts to relate among energy consumption, packet size and number of nodes. It is observed that in all the seven test scenarios the energy consumption increases for increase in data packet size as well as increase in the number of nodes with few exceptions which will be discussed below. Test I and Test III graph remains linear throughout, which means if the sensor nodes are ideally placed and connected, having only one forwarding node in their transmission range there will be no extra energy spent in duplicate transmission and overcoming congestion. Test II and Test VI plot remains linear up to data packet size of almost 400 bytes. This is a significant achievement on behalf of Test VI and reveals that Test VI can handle larger data packets reliably. It is also observed that all the seven graphs are linear up to 250 bytes data packet size. After this some of them drift. We observed this drift on per data packet basis in the simulator and it was revealed that the upward drift in energy consumption is due to congestion, duplicate transmission and other similar issue which require more energy. However the downward drift is due to packet loss, which was observed only in Test Rand after data packet size of 250 bytes. This shows that in Test Rand, packets are dropped after exceeding certain size limit and transmission becomes highly unreliable. Another interesting fact was observed in 3D deployment strategies, that the energy performance is better in our 3D node deployment strategy, Test VI, as compared to 3D grid deployment Test V, up to the packet size of 100 bytes (keeping in mind the number of nodes). After that the increase in energy consumption is observed in Test VI compared to Test V, due to increased number of nodes in Test VI. This is apparent up to 350 bytes packet size. After that Test VI consumes more energy and the graph becomes non-linear. This means that for data packet size more than 350 bytes the 3D grid set-up, Test V performs better in energy expenditure. Which means for non-critical, monitoring applications where reliability is not a big issue Test V set-up will perform better for larger data traffic. An efficient mac protocol can overcome this in Test VI by performing better data scheduling to over come the energy expense for larger data packet size.

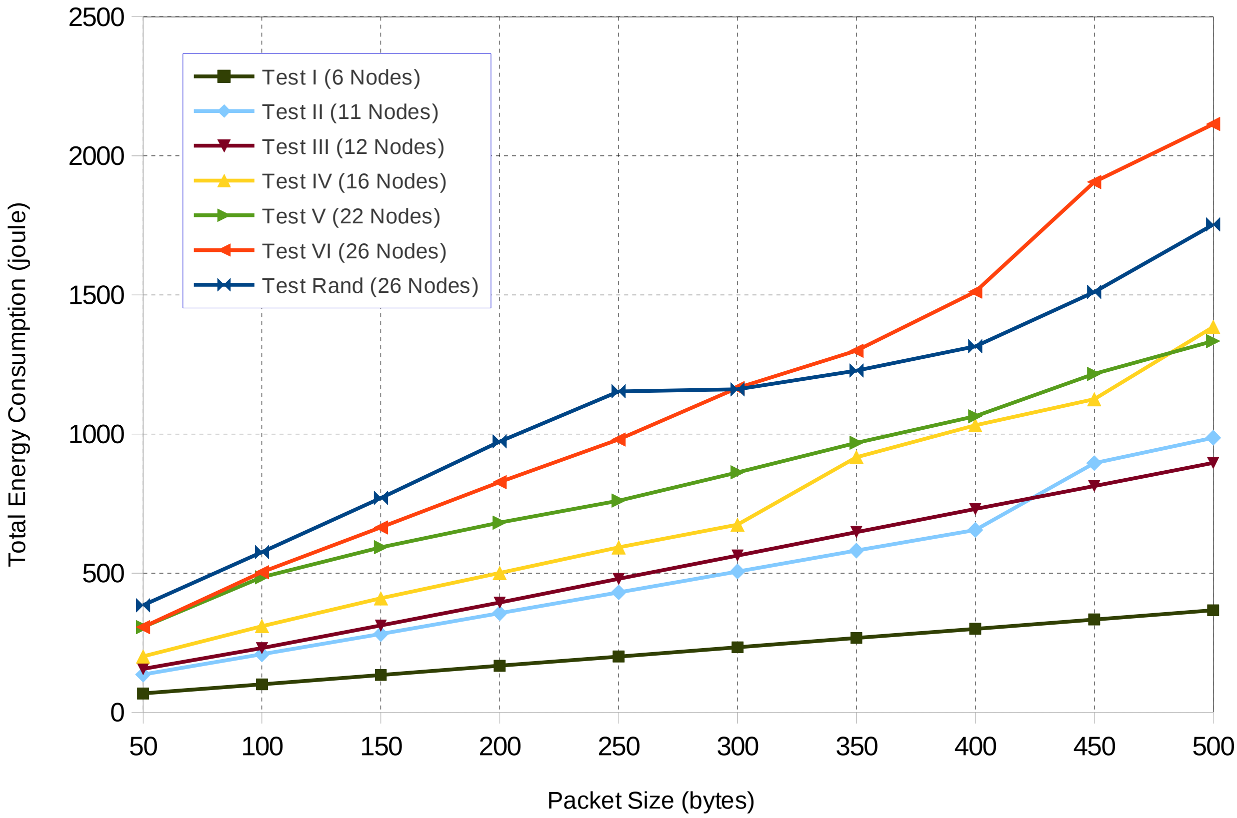


Figure 6.2: Total Energy Consumption Comparison of all Test set-ups

However, in comparison to Test Rand, random 3D deployment, our Test VI performs much better for the same number of sensor nodes till the data packet size 250 bytes. After this the data transmission in Test Rand becomes unstable due to packet loss which is shown as low energy consumption in the graph. Therefore with random 3D deployment we compromise on energy and for larger packet size, on reliable transmission as well. The simulation results also revealed that in Test Rand some of the nodes drain more energy compared to others hence the energy load distribution is not properly managed which will affect the network lifetime adversely. In Figure 6.3, the 3D comparison plot also reveals these results for all the test set-ups.

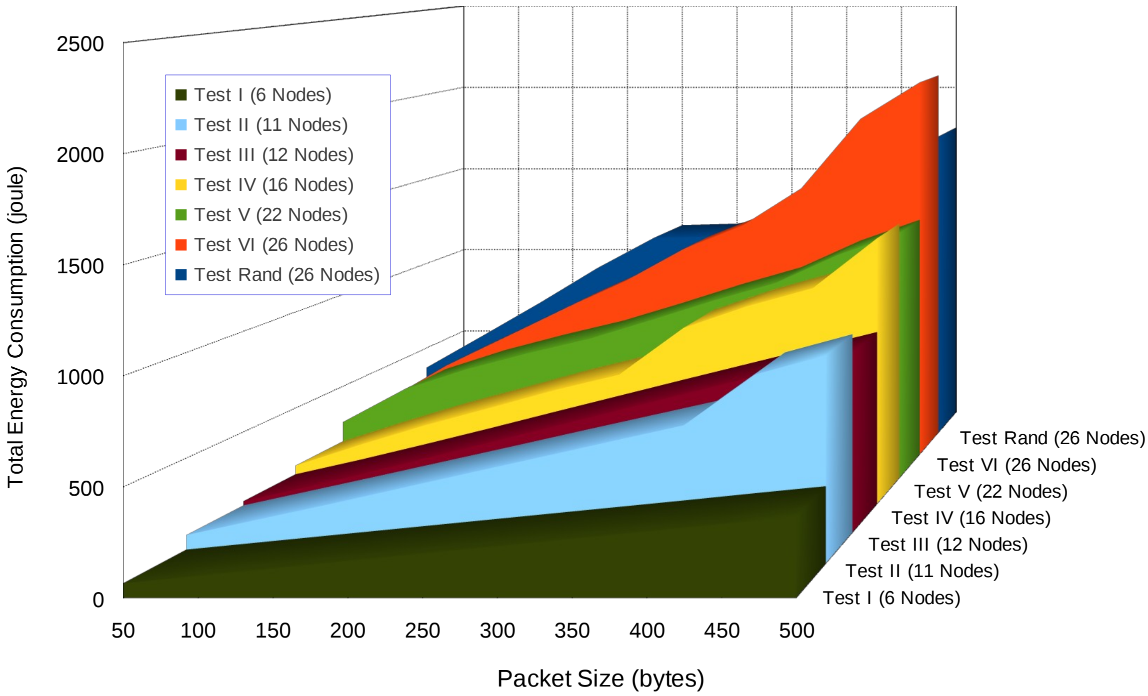


Figure 6.3: Total Energy Consumption Comparison of all Test set-ups in 3D

6.3 Energy Consumption per Node

The energy consumption per node versus data packet size comparison of all the test set-ups elaborates some very interesting findings. These plots are overlapping so it is easier to observe them in the form of a bar chart as shown in Figure 6.4. One of the interesting facts is that our 3D node deployment strategy, Test VI performs as efficient as 2D strategies (Test I to Test IV) and better than the rest of 3D set-ups, up to the data packet size of 150 bytes. After that Test V takes up the lead in 3D deployment strategies. This means that Test VI still performs comparable or better than most of the 2D set-ups however Test V performs even better. On the other hand Test Rand energy consumption per node is higher than any other test set-up till 250 bytes, after which it starts dropping the packets and effecting the overall reliability of the network communication as shown in Figure 6.4 and further elaborated in Figure 6.5.

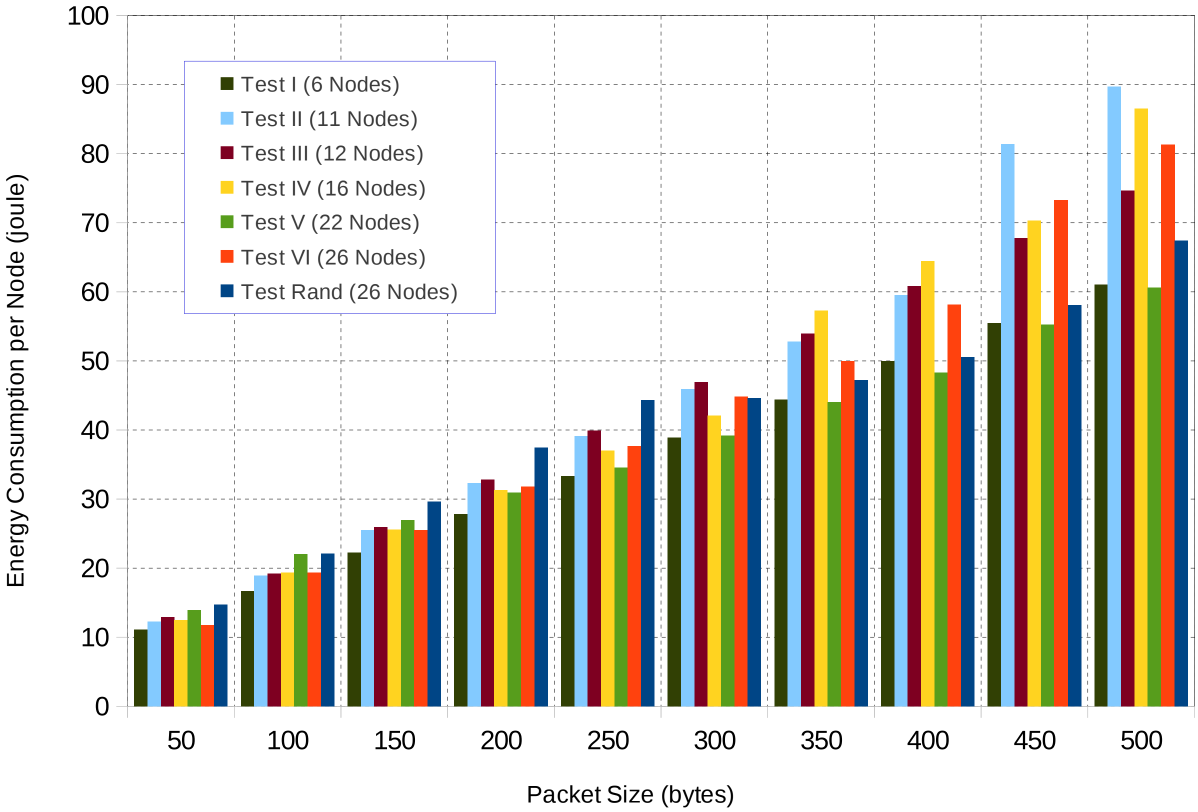


Figure 6.4: Energy Consumption per Node Analysis of all the Test set-ups

It is interesting to observe that initially Test VI, our 3D deployment strategy, performs better but after 150 bytes packet size Test V, which is a simple 3D cubic grid deployment gives better energy consumption performance per node. Hence for larger data packet size Test V performs slightly better than Test VI at the cost of reduction in back-up paths. We have elaborated this as a comparison between Test V, Test VI and Test Rand and comparing this comparison with the 2D tests in the same plot as shown in Figure 6.6.

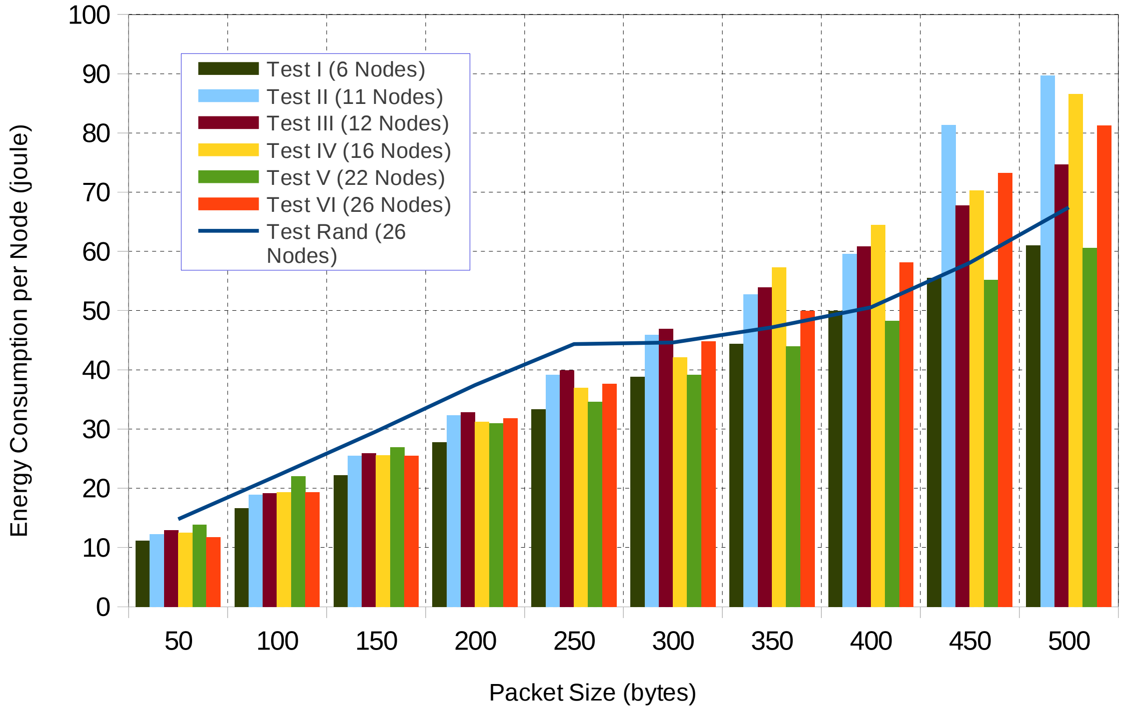


Figure 6.5: Energy Consumption per Node Analysis of all Test set-ups vs Test Rand

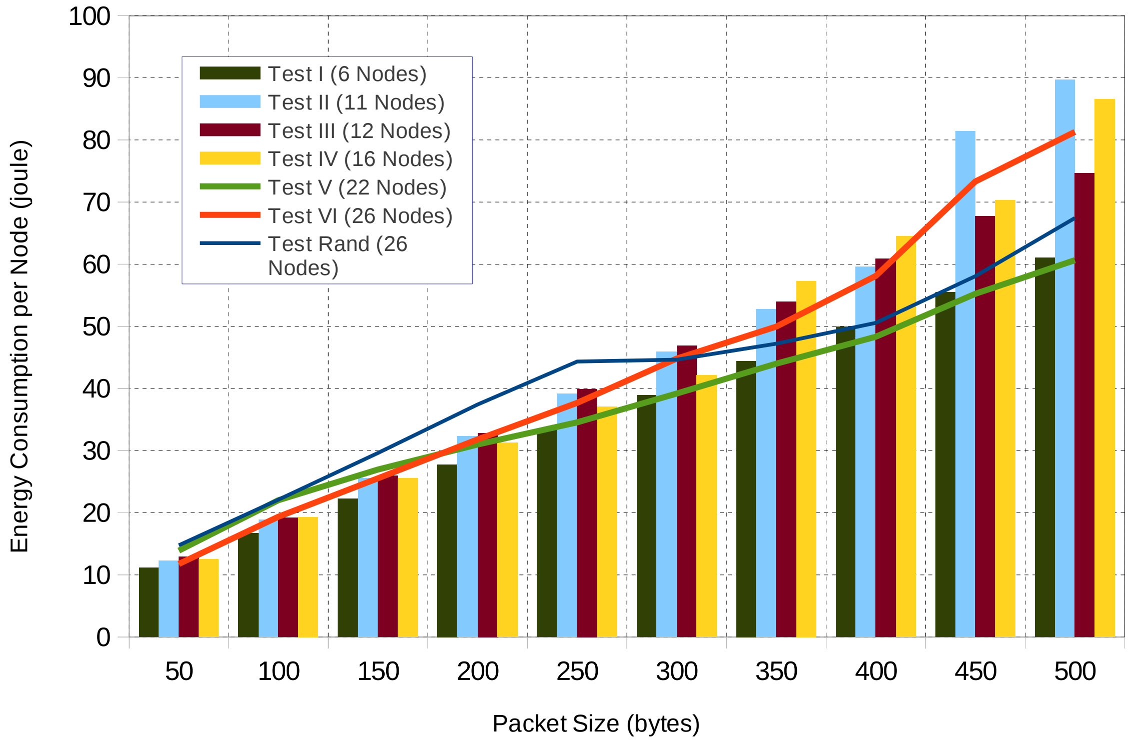


Figure 6.6: Energy Consumption per Node Analysis of 2D Test set-ups vs 3D Test set-ups

6.4 Total End-to-End Delay

In this section we analyse the total end-to-end delay of all the test set-ups. These plots are overlapping, so to observe them clearly we use bar chart as shown in Figure 6.7. It is obvious that Test Rand has the most end-to-end delay which is more than double as compared to other test set-ups. This is a clear evidence that Test Rand which is the 3D random deployment strategy highly compromises on the efficiency of data transmission.

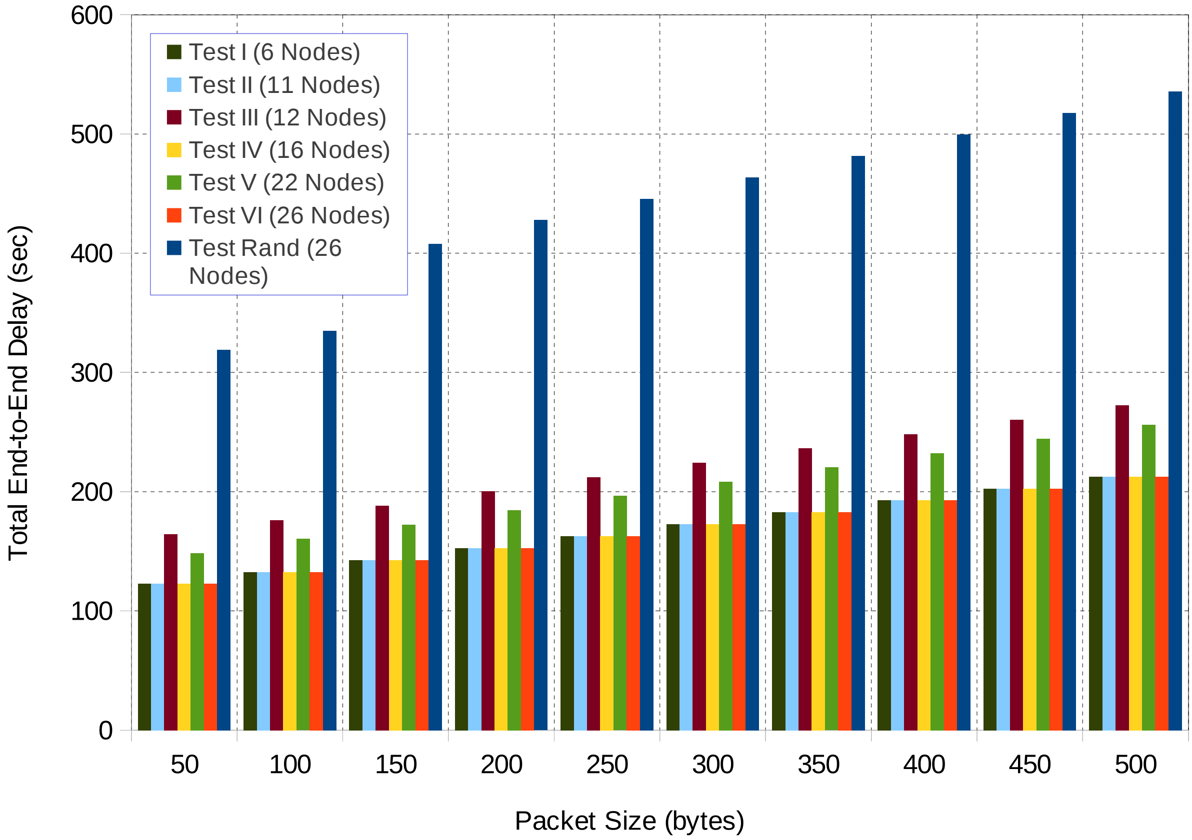


Figure 6.7: Total End-to-End Delay Analysis

Figure 6.8 shows the end-to-end delay comparison of 3D deployment strategies as compared to all of the 2D test set-ups. The end-to-end delay is greater in 2D grid deployment, Test III compared to Test IV. Similarly it is greater in 3D grid deployment, Test V compared to Test VI. This reveals an interesting fact that by adding the Reliability nodes, as in Test IV and Test VI the end-to-end delay decreases. Another interesting fact is that the end-to-end delay of our proposed design stays similar as in

Test I (which is straight vertical line of communication with optimum number of nodes for connectivity). Which means that in our 3D node deployment design (Test VI) we are not compromising at all at the efficiency of data transmission. This is a significant achievement as we are providing several back-up paths hence increasing the reliability of data transmission remarkably with no extra overall delay in communication.

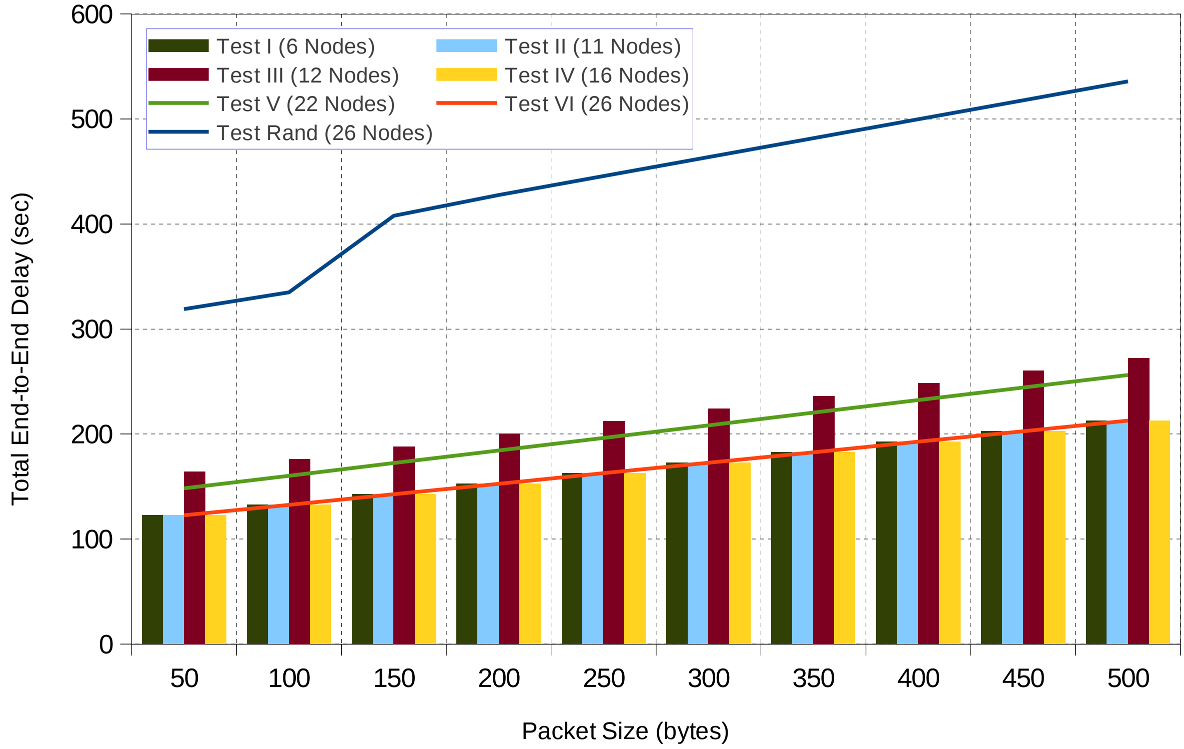


Figure 6.8: Total End-to-End Delay Analysis of 2D Test set-ups vs 3D Test set-ups

6.5 Delay per Node

In Figure 6.9 we compare the delay per node in all the test set-ups in order to build an idea how delay varies on per node basis in comparison to data packet size. Here we notice that in single vertical line of communication strategies i.e. Test I & Test II, least number of nodes which are in Test I (6 nodes) contribute to most delay, as overall travel distance is the same for less number of nodes. Therefore, the contribution of each node to the overall delay is more compared to other test set-ups. Further we see that increasing the number of nodes to almost double as in Test II (11 nodes), decreases the delay per node to almost half, which is expected. However, it is interesting to observe that delay per node reduces in 3D set-ups, compared to 2D, with least delay per node in our proposed test set-up, Test VI. Whereas the delay per node in Test Rand is far more than expected in a 3D set-up. This comparison is elaborated further in Figure 6.10 and 6.11.

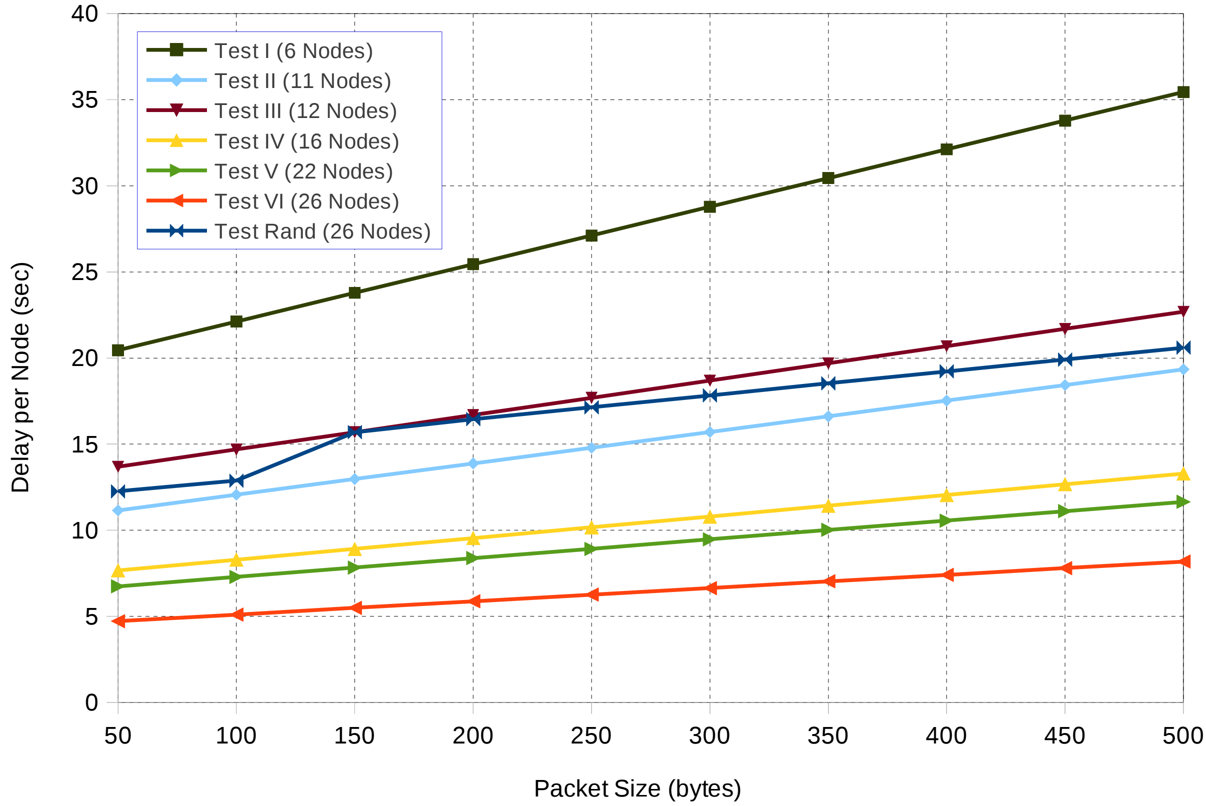


Figure 6.9: Delay per Node Analysis of all the Test set-ups

Another interesting fact we notice here is that delay per node contribution decreases by adding reliability nodes in both 2D and 3D set-ups. Which means that Reliability Nodes help in reducing the overall delay. All the test set-ups without Reliability Nodes have more delay per node. Therefore, we can conclude that by adding Reliability Nodes to a set-up we not only increase the back-up paths and reliable data transmission but also decrease the end-to-end delay contribution of each node.

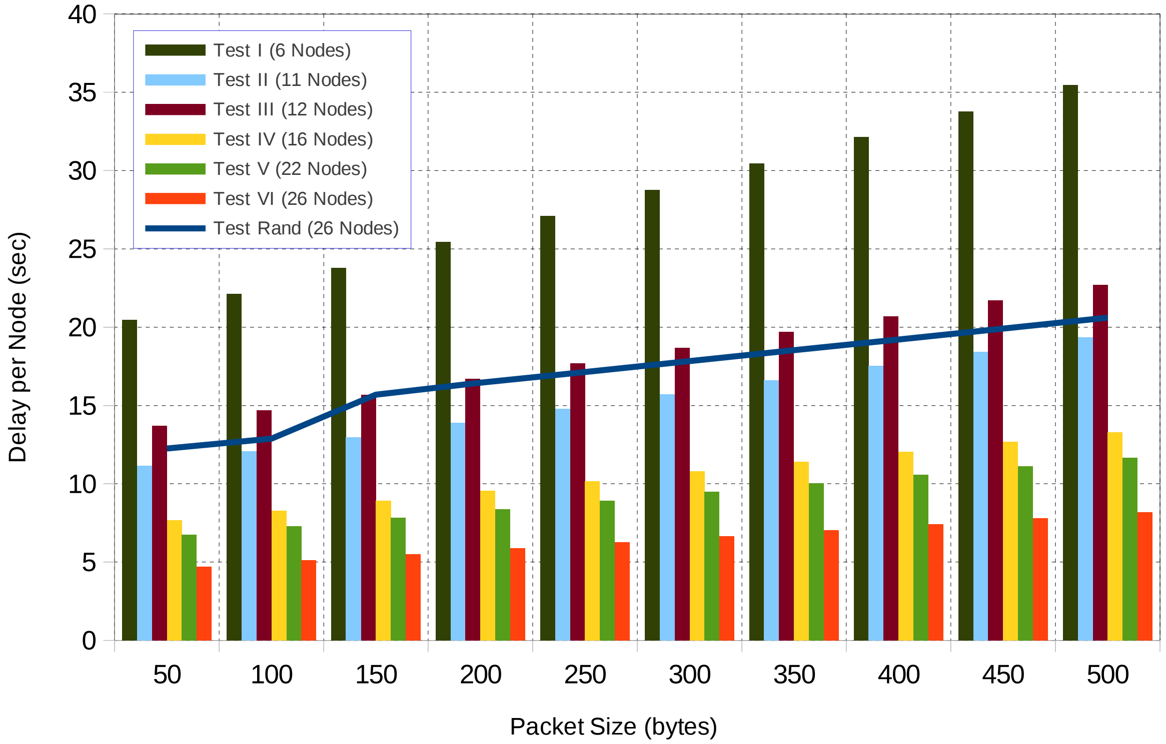


Figure 6.10: Delay per Node Analysis of all Test Set-ups vs Test Rand

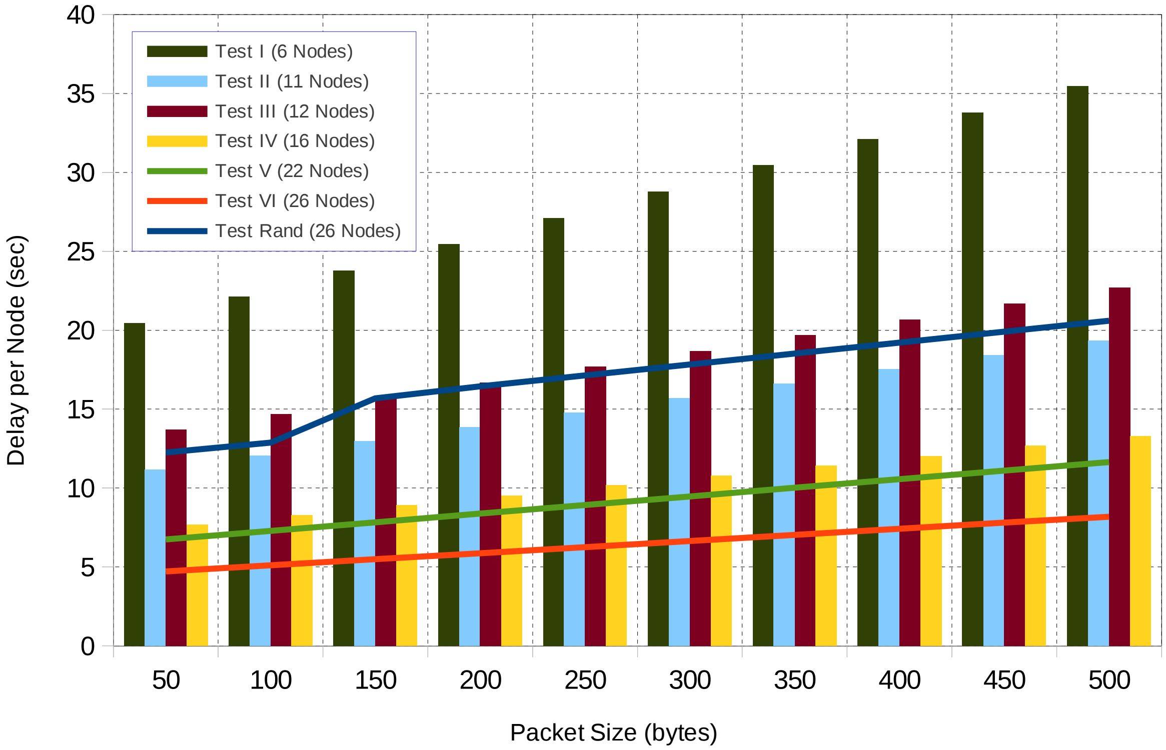


Figure 6.11: Delay per Node Analysis of 2D Test set-ups vs 3D Test set-ups

6.6 Conclusion

By comparing the results we can easily draw the conclusion that by deploying the sensor nodes randomly in a 3D underwater environment, there will be serious compromise in the optimum use of the resources (i.e. energy and time). This will not only increase the end-to-end delay it will increase the overall energy consumption of the network. The end-to-end delay will be adversely affected as the sensor nodes will not be distributed evenly and there will be certain regions where they will overlap each other's functioning zone (the sensing or transmitting sphere of a sensor node) hence causing congestion, delay and duplicate transmission. On the other hand there will be regions where only one sensor node will be handling all the data traffic (creating bottlenecks) in that region with no back-up node. These situations not only affect the end-to-end delay, they have a serious impact on network life time and reliability of data transmission. The energy consumption of the network is also affected as some nodes will be consuming more energy compared to others. This is due to the reason that they are not distributed evenly in the underwater volume. Although the overall energy consumption is not as much affected as delay because some of the nodes are not actively involved in transmission of data while other nodes are forwarding too much data traffic through them. From the energy consumption situation of the network we can easily infer that random 3D deployment will reduce the optimum network lifetime by allowing certain nodes to drain their batteries before others.

CHAPTER 7

Conclusion and Future Work

This chapter concludes the thesis by outlining our original contributions and providing directions for future research in this area. In this thesis, we present the techniques for improving the routing performance of an Underwater Wireless Sensor Network (UWSN). This has been achieved by testing a number of strategies (Test set-ups as discussed in Chapter 5) for node deployment in 3D underwater environment. T he effectiveness of each feature added from single line of vertical communication to the final 3D node deployment strategy, providing full sensing, communication and back-up coverage, is observed and analysed through several simulations. From the results of simulation experiments we proved that our 3D node deployment strategy not only provides reliability in data transmission but significantly improves routing efficiency in terms of energy consumption and end-to-end delay. The strategies proposed in this research work have been a result of careful consideration of the unique nature of underwater acoustic communication (such as high and variable propagation delay, low bandwidth capacity, high error probability and temporary loss of connectivity) as well as the specific characteristics of UWSN which makes it different from terrestrial WSN, such as 3D architecture and loss of connectivity due to node mobility etc.

7.1 Results and Contributions of the Research

In accordance with the motivations and objectives of our research, we improved the routing performance of an UWSN by designing a 3D node deployment strategy which also provides full sensing and communication coverage, reliability, decrease in delay and energy consumption. In this section, we summarize the original contributions of our research work.

7.1.1 Improved Routing Performance by 3D Node Deployment Strategy

- We designed an effective 3D node deployment strategy which offers full sensing and communication coverage.
- The concept is based on the strong anisotropic nature of the acoustic signal propagation in underwater environment whereby the multipath spread in horizontal channels is extremely long compared to vertical channels.
- We designed a system to combat the effect of multipath to achieve high data throughput and improve the routing performance.
- We based our design on the vertical transmission, highlighting the fact that in UWSN, the data needs to be transmitted from seabed to sea surface and not in a horizontal direction as in terrestrial WSN.
- We analysed several test scenarios through simulation experiments and verified that our design not only provides full sensing and communication coverage, it improves the overall routing performance significantly in terms of energy consumption and end-to-end delay.
- This strategy can be implemented to any existing routing protocol to improve its performance and reliability.

7.1.2 Providing Full Coverage and Connectivity

Another significant achievement of our design is that it provides full coverage and connectivity with several back-up paths. Although this is one of the key objectives of a sensor network but very difficult to achieve practically in an underwater environment due to the changing network configuration and challenging wireless acoustic communication. There are routing strategies which deal with the creation of void (a region created due to node movement or failure, which is not covered by any other node) by bypassing the void and re-routing the data around it towards the sink. However in this way, any activity happening in the void region is completely ignored. In critical monitoring applications, transmitting data from selective regions can cause huge loss. Our 3D node deployment strategy successfully address this issue by providing a full coverage to the sensed region.

7.1.3 Data Load and Energy Consumption Distribution

In an underwater environment, the dynamic network configuration due to water current, node movement or node failure, results in an uneven distribution of sensor nodes, causing loss of connectivity and uneven distribution of data load. In our design, in addition to main forwarding paths there are several back-up paths automatically generated. These back-up paths play a vital role in the distribution of data as well as energy load. This is achieved at no extra cost to the resources. These back-up paths not only ensure sensing as well as transmission coverage in a 3D environment, it will distribute the data and energy load within the network. Hence certain regions will not drain batteries sooner than others, which usually happens in random node deployment. Distribution of data and energy load resulting from our design will prolong the network life time.

7.1.4 Providing Reliability

Most of the routing protocols do not offer sufficient reliability for the transmission of data in sparse 3D UWSN or address the changing network configuration due to node movement. Our strategy overcomes these issues by providing reliable data communication, which is one of the prime focus of our design. With all the sensor nodes being well connected, having several backup paths we ensure reliable sensing as well as transmission of data in our design. While developing this strategy step by step, we observed through simulation experiments that certain nodes placement in the overall design improved the reliability and routing performance significantly. Hence we called these nodes as “Reliability Nodes” in our design in Chapter 5, due to their sole purpose of providing reliability and increasing the network lifetime without causing any extra resource consumption. Even by increasing the data traffic, our 3D node deployment strategy ensures reliable transmission with no loss of data due to several back-up paths available which is in contrast to random deployment where the packet loss occurs by increasing the data traffic. These back-up routes can also combat the dynamic nature of UWSN due to water current and node movement.

7.1.5 Decrease in End-to-End Delay

Whilst improving the routing protocol performance, we achieved a significant decrease in end-to-end delay, especially after adding the Reliability nodes to our 3D node deployment design. This is particularly important achievement in UWSN, where due to acoustic communication, the speed of transmission of data is 5 orders of magnitude less than in radio communication. In addition to this, the sensor networks and especially underwater sensor networks always strive for real-time data transmission. Hence this is a remarkable achievement. Our 3D node deployment design improves the transmission time significantly by decreasing the end-to-end delay for the transmission of data from source node to the destination node as shown in the simulation results in Chapter 6.

7.2 Future Work

This section highlights areas for future research work in this direction based on the results and contribution of this thesis work.

7.2.1 Implementation of the Strategy in Field on Real Hardware

Performing underwater wireless sensor networking experiments in the field on real hardware are very expensive. However, it will be worth implementing this 3D node deployment strategy on real hardware, to test it in the field practically. And then compare the results with the simulations. Underwater channel propagation is very unpredictable, it will be worth observing the results in different parts of the sea.

7.2.2 Testing of the Strategy on a Testbed

To avoid the expensive nature of underwater wireless sensor network experimentation in the real sea, and the simplicity of physical layer designed in some simulators, there is an intermediate solution, a step between simulation and at sea experimentation which is the testbed. Some of the testbeds are discussed in chapter 3. It will be a great opportunity to implement and test this strategy on a testbed to analyse its performance using real hardware.

7.2.3 Utilizing the Strategy for Localization

Efficient localization in UWSN is an on going issue. Our 3D node deployment strategy presented in this research work can be used as a baseline scheme to develop an efficient localization algorithm on top for UWSN. As this strategy ensures full sensing and communication coverage which can act as a backbone for any localization scheme. This strategy is simple to implement and provides full coverage to the monitored 3D underwater environment.

7.2.4 Experimenting with other Routing protocols

This strategy promotes modular design approach, it can be integrated with any other routing protocol to improve its performance, as it is independent of any changes to the routing protocol or vice versa. We have tested it with Vector Based Forwarding Protocol (VBF) [91] in this thesis work but its scope is not limited to or dependent on VBF. It can be implemented to other routing protocols independently in a similar way to improve the output efficiency.

7.2.5 Testing other 3D Lattice Structures

In this thesis work we implemented our 3D Node deployment strategy to cubic lattice structure. The main concept of this strategy, vertical communication in UWSN and providing several back-up paths by intelligently placing reliability nodes to combat the dynamic nature of UWSN, can be implemented on other shapes like hexagonal prism deployment, rhombic dodecahedron deployment and truncated octahedron deployment etc. In general any 3D node deployment configuration can benefit from the significant reliability added by intelligent placement of Reliability Nodes in the middle of each cell of the 3D lattice structure. It will be worth trying this strategy on other shapes to see which one will be the most efficient.

7.3 Final Remarks

Underwater wireless sensor networks are considered to be an extension of wireless sensor networks. It may be true conceptually as both are based on the same technology but practically the underwater environment compels the underwater wireless sensor networking technology to be surrounded by completely different challenges than WSN technology. For example acoustic communication, which is the main physical layer technology in UWSN, suffers from slow propagation speed, low bandwidth capacity and high error probability compared to radio communication in WSN. Most importantly the 3D underwater environment and water current makes the UWSN behave completely different from WSN. These very basic differences in both technologies make them completely different from each other.

In past, most of the networking design concepts and node deployment strategies are adopted from terrestrial WSN and redesigned with minor changes to fit in the underwater environment. However, due to the fundamental differences between the UWSN and WSN technology, these designs are not suitable for 3D underwater environment. 3D network design which is a prime requirement of an UWSN, is much more complicated than 2D. Most of the networking protocols developed for UWSN are tested in 2D with random node deployment and the results are assumed to be acceptable to the actual 3D underwater environment. First of all keeping in mind the size, sensing range and communication range of UWSN node, random node deployment will cause major waste of resources. Secondly, with random node deployment in 3D, there will be high probability of creation of void as well as congestion regions. Void is like a vacuum created in a sensor network, due to absence of a sensor node, having no coverage or connectivity. This is caused by random deployment, node failure or node movement due to water current. On the other hand, congestion regions are the regions covered by too many nodes, hence causing congestion and other related issues. It is clear that random node deployment in 3D cannot guarantee the optimum use of resources to offer reliability, which can be offered to the network if the sensor nodes are properly deployed in a certain manner.

In this thesis work we proposed a 3D node deployment strategy, after careful consideration of the unique characteristics of underwater acoustic communication and specific features of UWSN which makes it different from terrestrial WSN, such as 3D architecture and loss of connectivity due to node mobility etc. This strategy purely targets 3D UWSN and not only provides full coverage and connectivity for data communication, it improves the routing protocol performance significantly in terms of energy consumption and especially end to end delay. We developed this strategy step by step from a single line of vertical communication to an effective 3D node deployment for UWSN. We performed several simulation experiments after adding each feature to the final design to observe the impact on the overall routing performance. We observed the impact of adding Reliability Nodes in our design, which were named after offering significant reliability and increasing the network lifetime by improving the routing performance. Finally, we verified that our design provides reliability to the UWSN at no extra cost to the resources.

Then we compared our results to random node deployment in 3D. This is commonly used for analysing the performance of UWSN routing protocols. The comparison results verified our effective deployment design and showed that it provides almost 25% less energy consumption to the random deployment and almost 150% less end-to-end delay. It also revealed that by increasing the data traffic, our 3D node deployment strategy has no loss of data due to several back-up paths available, which is in contrast to random deployment, where packet loss occurs by increasing the data traffic. Underwater environment is highly unpredictable, improving the routing performance by carefully analysing the impact of 3D node deployment strategy, ensuring full sensing, transmission and back-up coverage is a novel approach. This embedded with any networking protocol will improve its performance significantly.

References

[1] M. Stojanovic, "Underwater acoustic communication," in *Wiley Encyclopedia of Electrical and Electronics Engineering,* John Wiley & Sons, Inc., 2001, .

[2] T. Melodia, H. Kulhandjian, L. Kuo and E. Demirors, "Advances in underwater acoustic networking," *Mobile Ad Hoc Networking: Cutting Edge Directions,* vol. 852, 2013.

[3] U. M. Cella, R. Johnstone and N. Shuley, "Electromagnetic wave wireless communication in shallow water coastal environment: Theoretical analysis and experimental results," in *Proceedings of the Fourth ACM International Workshop on UnderWater Networks,* Berkeley, California, 2009, pp. 9:1-9:8.

[4] N. Farr, A. Bowen, J. Ware, C. Pontbriand and M. Tivey, "An integrated, underwater optical /acoustic communications system," in *OCEANS 2010 IEEE - Sydney,* 2010, pp. 1-6.

[5] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin and P. Corke, "Data collection, storage, and retrieval with an underwater sensor network," in *Proceedings of the 3rd International Conference on Embedded* Networked *Sensor Systems,* San Diego, California, USA, 2005, pp. 154-165.

[6] L. Liu, S. Zhou and J. -. Cui, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Communications and Mobile Computing,* vol. 8, pp. 977-994, 2008.

[7] F. Hu and X. Cao, *Wireless Sensor Networks : Principles and Practice.* Boca Raton, FL: CRC Press, 2010.

[8] M. C. Domingo and R. Prior, "Energy analysis of routing protocols for underwater wireless sensor networks," *Comput. Commun.,* vol. 31, pp. 1227-1238, 2008.

[9] I. F. Akyildiz, D. Pompili and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks,* vol. 3, pp. 257-279, 2005.

[10] (January, 2016). *Autonomous Ocean Sampling Network (AOSN), Monterey Bay Aquarium Research Institute (MBARI)* [Autonomous Ocean Sampling Network (AOSN), Monterey Bay Aquarium Research Institute (MBARI))]. Available: http://www3.mbari.org/aosn/.

[11] A. Khan and L. Jenkins, "Undersea wireless sensor network for ocean pollution prevention," in *Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on,* 2008, pp. 2-8.

[12] G. Acar and A. Adams, "ACMENet: an underwater acoustic sensor network protocol for real-time environmental monitoring in coastal areas," *IEE Proceedings-Radar, Sonar and Navigation,* vol. 153, pp. 365-380, 2006.

[13] J. Heidemann, W. Ye, J. Wills, A. Syed and Y. Li, "Research challenges and applications for underwater sensor networking," in *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE,* 2006, pp. 228-235.

[14] P. Kumar, P. Kumar and P. Priyadarshini, "Underwater acoustic sensor network for early warning generation," in *Oceans, 2012,* 2012, pp. 1-6.

[15] K. Casey, A. Lim and G. Dozier, "A sensor network architecture for tsunami detection and response," *International Journal of Distributed Sensor Networks,* vol. 4, pp. 28-43, 2008.

[16] J. Lloret, "Underwater sensor nodes and networks," *Sensors,* vol. 13, pp. 11782-11796, 2013.

[17] H. Denkilkian, A. Koulakezian, R. Ohannessian, M. S. Chalfoun, M. K. W. Joujou, A. Chehab and I. H. Elhajj, "Wireless sensor for continuous real-time oil spill thickness and location measurement," *Instrumentation and Measurement, IEEE Transactions on,* vol. 58, pp. 4001-4011, 2009.

[18] C. Alippi, R. Camplani, C. Galperti and M. Roveri, "A robust, adaptive, solar-powered WSN framework for aquatic environmental monitoring," *Sensors Journal, IEEE,* vol. 11, pp. 45-55, 2011.

[19] C. A. Pérez, M. Jimenez, F. Soto, R. Torres, J. López and A. Iborra, "A system for monitoring marine environments based on wireless sensor networks," in *OCEANS, 2011 IEEE-Spain,* 2011, pp. 1-6.

[20] 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, pp. 373-390, 2012.

[21] H. Yang, H. Wu and Y. He, "Architecture of wireless sensor network for monitoring aquatic environment of marine shellfish," in *Asian Control Conference, 2009. ASCC 2009. 7th,* 2009, pp. 1147-1151.

[22] S. Bainbridge, D. Eggeling and G. Page, "Lessons from the field—Two years of deploying operational wireless sensor networks on the Great Barrier Reef," *Sensors,* vol. 11, pp. 6842-6855, 2011.

[23] J. Trevathan, R. Johnstone, T. Chiffings, I. Atkinson, N. Bergmann, W. Read, S. Theiss, T. Myers and T. Stevens, "SEMAT—the next generation of inexpensive

marine environmental monitoring and measurement systems," *Sensors*, vol. 12, pp. 9711-9748, 2012.

[24] Peng, Chao, Xinguo Zhao, and Guangxu Liu. "Noise in the sea and its impacts on marine organisms." *International journal of environmental research and public health* 12, no. 10: 12304-12323, 2015.

[25] Harris, P., Philip, R., Robinson, S. and Wang, L. “Monitoring Anthropogenic Ocean Sound from Shipping Using an Acoustic Sensor Network and a Compressive Sensing Approach, ” *Sensors*, vol. 3, pp. 415, 2016.

[26] N. Mohamed, I. Jawhar, J. Al-Jaroodi and L. Zhang, "Sensor network architectures for monitoring underwater pipelines," *Sensors*, vol. 11, pp. 10738-10764, 2011.

[27] G. Han, A. Qian, C. Zhang, Y. Wang and J. J. Rodrigues, "Localization algorithms in large-scale underwater acoustic sensor networks: a quantitative comparison," *International Journal of Distributed Sensor Networks,* vol. 2014, 2014.

[28] M. Waldmeyer, H. Tan and W. K. Seah, "Multi-stage AUV-aided localization for underwater wireless sensor networks," in *Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on,* 2011, pp. 908-913.

[29] Y. Guo and Y. Liu, "Localization for anchor-free underwater sensor networks," *Comput. Electr. Eng.,* vol. 39, pp. 1812-1821, 2013.

[30] P. Carroll, S. Zhou, K. Mahmood, H. Zhou, X. Xu and J. Cui, "On-demand asynchronous localization for underwater sensor networks," in *Oceans, 2012,* 2012, pp. 1-4.

[31] S. Kemna, M. J. Hamilton, D. T. Hughes and K. D. LePage, "Adaptive autonomous underwater vehicles for littoral surveillance," *Intelligent Service Robotics,* vol. 4, pp. 245-258, 2011.

[32] A. Caiti, V. Calabro, A. Munafò, G. Dini and A. Lo Duca, "Mobile underwater sensor networks for protection and security: field experience at the UAN11 experiment," *Journal of Field Robotics,* vol. 30, pp. 237-253, 2013.

[33] E. Cayirci, H. Tezcan, Y. Dogan and V. Coskun, "Wireless sensor networks for underwater survelliance systems," *Ad Hoc Networks,* vol. 4, pp. 431-446, 2006.

[34] T. Kanazawa, "Japan trench earthquake and tsunami monitoring network of cable-linked 150 ocean bottom observatories and its impact to earth disaster science," in *Underwater Technology Symposium (UT), 2013 IEEE International,* 2013, pp. 1-5.

[35] B. Ayaz, A. Allen and M. Wiercigroch, "Dynamically reconfigurable routing protocol design for underwater wireless sensor network," in *International*

Conference on Sensing Technology, 2014, .

[36] J. Kim, N. Yun, S. Muminov, S. Park and O. Yi, "Security in Underwater Acoustic Sensor Network: Focus on Suitable Encryption Mechanisms," pp. 160-168, 2012.

[37] Y. Cong, G. Yang, Z. Wei and W. Zhou, "Security in underwater sensor network," in *Communications and Mobile Computing (CMC), 2010 International Conference on,* 2010, pp. 162-168.

[38] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *Communications Magazine, IEEE,* vol. 47, pp. 84-89, 2009.

[39] X. Lurton, *An Introduction to Underwater Acoustics: Principles and Applications.* Springer, 2010.

[40] M. A. Ainslie and J. G. McColm. A simplified formula for viscous and chemical absorption in sea water. *J. Acoust. Soc. Am. 103(3),* pp. 1671-1672. 1998. . DOI: http://dx.doi.org/10.1121/1.421258.

[41] L. Brekhovskikh and Y. Lysanov, *Fundamentals of Ocean Acoustics.* Springer Berlin Heidelberg, 1982.

[42] C. A. Balanis, *Advanced Engineering Electromagnetics.* John Wiley & Sons, 1989.

[43] J. H. Goh, A. Shaw and A. Al-Shamma'a, "Underwater wireless communication system," in *Journal of Physics: Conference Series,* 2009, pp. 012029.

[44] A. I. Al-Shamma'a, A. Shaw and S. Saman, "Propagation of electromagnetic waves at MHz frequencies through seawater," *Antennas and Propagation, IEEE Transactions on,* vol. 52, pp. 2843-2849, 2004.

[45] J. Mulholland and D. McStay, "Wireless communication enhances subsea production monitoring," *Offshore Magazine,* vol. 71, 2011.

[46] M. Kerker, *The Scattering of Light and Other Electromagnetic Radiation.* New York: Academic Press, 1973.

[47] J. D. Zhang, Y. W. Liu, S. M. Gao, C. Z. Li, F. Zhang, H. M. Zen and C. S. Ye, "Pilot testing of outside-in UF pretreatment prior to RO for high turbidity seawater desalination," *Desalination,* vol. 189, pp. 269-277, 3/1, 2006.

[48] M. Stojanovic, "Recent advances in high-speed underwater acoustic communications," *Oceanic Engineering, IEEE Journal of,* vol. 21, pp. 125-136, 1996.

[49] M. Stojanovic, J. A. Catipovic and J. G. Proakis. Reduced-complexity spatial and temporal processing of underwater acoustic communication signals. *J.*

Acoust. Soc. Am. 98(2), pp. 961-972. 1995. DOI: http://dx.doi.org/10.1121/1.413521.

[50] 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, pp. 795, 2014.

[51] A. A. Syed, W. Ye, J. Heidemann and B. Krishnamachari, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proceedings of the Second Workshop on Underwater Networks,* Montreal, Quebec, Canada, 2007, pp. 41-48.

[52] J. Rice, B. Creber, C. Fletcher, P. Baxley, K. Rogers, K. McDonald, D. Rees, M. Wolf, S. Merriam, R. Mehio, J. Proakis, K. Scussel, D. Porta, J. Baker, J. Hardiman and D. Green, "Evolution of seaweb underwater acoustic networking," in *OCEANS 2000 MTS/IEEE Conference and Exhibition,* 2000, pp. 2007-2017 vol.3.

[53] L. Kleinrock and F. A. Tobagi, "Packet Switching in Radio Channels: Part I--Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *Communications, IEEE Transactions on,* vol. 23, pp. 1400-1416, 1975.

[54] N. Chirdchoo, Wee-Seng Soh and Kee Chaing Chua, "Aloha-based MAC protocols with collision avoidance for underwater acoustic networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE,* 2007, pp. 2271-2275.

[55] M. Molins and M. Stojanovic, "Slotted FAMA: A MAC protocol for underwater acoustic networks," in *OCEANS 2006 - Asia Pacific,* 2006, pp. 1-7.

[56] C. Petrioli, R. Petroccia and J. Potter, "Performance evaluation of underwater MAC protocols: From simulation to at-sea testing," in *OCEANS, 2011 IEEE - Spain,* 2011, pp. 1-10.

[57] D. Pompili, T. Melodia and I. F. Akyildiz, "A CDMA-based Medium Access Control for UnderWater Acoustic Sensor Networks," *Wireless Communications, IEEE Transactions on,* vol. 8, pp. 1899-1909, 2009.

[58] L. Kuo and T. Melodia, "Distributed Medium Access Control Strategies for MIMO Underwater Acoustic Networking," *Wireless Communications, IEEE Transactions on,* vol. 10, pp. 2523-2533, 2011.

[59] N. Nicolaou, A. See, Peng Xie, Jun-Hong Cui and D. Maggiorini, "Improving the robustness of location-based routing for underwater sensor networks," in *OCEANS 2007 - Europe,* 2007, pp. 1-6.

[60] 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 Underwater Networks,* 2008, pp. 75-82.

[61] H. Yan, Z. J. Shi and J. Cui, "DBR: Depth-based routing for underwater sensor networks," in *NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet,* Springer, 2008, pp. 72-86.

[62] M. Ayaz and A. Abdullah, "Hop-by-hop dynamic addressing based (H2-DAB) routing protocol for underwater wireless sensor networks," in *Information and Multimedia Technology, 2009. ICIMT'09. International Conference on,* 2009, pp. 436-441.

[63] N. Chirdchoo, W. Soh and K. C. Chua, "Sector-based routing with destination location prediction for underwater mobile networks," in *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on,* 2009, pp. 1148-1153.

[64] A. Wahid and K. Dongkyun, “Analyzing routing protocols for underwater wireless sensor networks,” *International Journal of Communication Networks and Information Security Ijcnis 2(3),* 2010.

[65] W. K. Seah and H. Tan, "Multipath virtual sink architecture for underwater sensor networks," in *OCEANS 2006-Asia Pacific,* 2007, pp. 1-6.

[66] D. Pompili, T. Melodia and I. F. Akyildiz, "A resilient routing algorithm for long-term applications in underwater sensor networks," in *Proc. of Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net),* 2006, .

[67] J. Chen, X. Wu and G. Chen, "REBAR: A reliable and energy balanced routing algorithm for UWSNs," in *Grid and Cooperative Computing, 2008. GCC'08. Seventh International Conference on,* 2008, pp. 349-355.

[68] A. Wahid and K. Dongkyun, "Analyzing routing protocols for underwater wireless sensor networks," *International Journal of Communication Networks and Information Security,* vol. 2, pp. 253, 2010.

[69] M. C. Domingo, "A distributed energy-aware routing protocol for underwater wireless sensor networks," *Wireless Personal Communications,* vol. 57, pp. 607-627, 2011.

[70] U. Lee, P. Wang, Y. Noh, L. F. M. Vieira, M. Gerla and J. Cui, "Pressure routing for underwater sensor networks." in *Infocom,* 2010, pp. 1676-1684.

[71] M. Ayaz, A. Abdullah and L. T. Jung, "Temporary cluster based routing for underwater wireless sensor networks," in *Information Technology (ITSim), 2010 International Symposium in,* 2010, pp. 1009-1014.

[72] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J. Cui, Z. Shi, Y. Fei and S. Zhou, "Aqua-Sim: An NS2 based simulator for underwater sensor networks," 2009.

[73] F. Guerra, P. Casari and M. Zorzi, "World ocean simulation system (WOSS): A simulation tool for underwater networks with realistic propagation modeling," in *Proceedings of the Fourth ACM International Workshop on UnderWater*

Networks, Berkeley, California, 2009, pp. 4:1-4:8.

[74] M. Porter et al., “Bellhop code.” [Online]. Available:http://oalib.hlsresearch.com/Rays/index.html

[75] M. B. Porter, "The bellhop manual and user’s guide: Preliminary draft." *Heat, Light, and Sound Research, Inc., La Jolla, CA, USA, Tech. Rep*, 2011.

[76] O. C Rodriguez, "General description of the BELLHOP ray tracing program." *oalib. hlsresearch. com/Rays/GeneralDescription. pdf [Accessed: Jan. 2012]*, 2008.

[77] P. Casari, C. Tapparello, F. Guerra, F. Favaro, I. Calabrese, G. Toso, S. Azad, R. Masiero and M. Zorzi, "Open source suites for underwater networking: WOSS and DESERT underwater," *Network, IEEE*, vol. 28, pp. 38-46, 2014.

[78] J. Heidemann, M. Stojanovic and M. Zorzi. Underwater sensor networks: Applications, advances and challenges. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 370(1958)*, pp. 158-175. 2011. . DOI: 10.1098/rsta.2011.0214.

[79] R. Masiero, S. Azad, F. Favaro, M. Petrani, G. Toso, F. Guerra, P. Casari and M. Zorzi, "DESERT underwater: An NS-miracle-based framework to design, simulate, emulate and realize test-beds for underwater network protocols," in *OCEANS, 2012 - Yeosu*, 2012, pp. 1-10.

[80] C. Petrioli, R. Petroccia, J. R. Potter and D. Spaccini, "The SUNSET framework for simulation, emulation and at-sea testing of underwater wireless sensor networks," *Ad Hoc Networks*, 2014.

[81] R. Petroccia and D. Spaccini, "A back-seat driver for remote control of experiments in underwater acoustic sensor networks," in *OCEANS - Bergen, 2013 MTS/IEEE*, 2013, pp. 1-9.

[82] M. Prats, J. Perez, J. J. Fernandez and P. J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 2577-2582.

[83] J. M. Jornet, "AUVNetSim: A simulator for underwater acoustics networks," MIT Sea Grant, 2008.

[84] J. Llor and M. P. Malumbres, "Underwater Wireless Sensor Networks: How Do Acoustic Propagation Models Impact the Performance of Higher-Level Protocols" *Sensors*, vol. 12, pp. 1312, 2012.

[85] (Wednesday, September 09, 2015). *S&T Organization, Centre for Maritime Research and Experimentation (CMRE)*. Available: http://www.cmre.nato.int/.

[86] D. Hughes, R. Been and A. Vermeij, "Heterogeneous underwater networks for

ASW: technology and techniques," *Maritime Systems and Technology, Genova, Italy,* 2008.

[87] J. Cui, S. Zhou, Z. Shi, J. O'Donnell, Z. Peng, M. Gerla, B. Baschek, S. Roy, P. Arabshahi and X. Zhang, "Ocean-TUNE: A community ocean testbed for underwater wireless networks," in *Proceedings of the Seventh ACM International Conference on Underwater Networks and Systems*, Los Angeles, California, 2012, pp. 17:1-17:2.

[88] Z. Peng, J. Cui, B. Wang, K. Ball and L. Freitag, "An underwater network testbed: Design, implementation and measurement," in *Proceedings of the Second Workshop on Underwater Networks,* Montreal, Quebec, Canada, 2007, pp. 65-72.

[89] M. Ayaz, I. Baig, A. Abdullah and I. Faye, "A survey on routing techniques in underwater wireless sensor networks," *Journal of Network and Computer Applications*, vol. 34, pp. 1908-1927, 2011.

[90] N. Li, J. Martínez, J. M. Meneses Chaus and M. Eckert, "A Survey on Underwater Acoustic Sensor Network Routing Protocols," *Sensors,* vol. 16, pp. 414, 2016.

[91] P. Xie, Z. Zhou, N. Nicolaou, A. See, J. Cui and Z. Shi, "Efficient vector-based forwarding for underwater sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, pp. 1-13, 2010.

[92] Chang Su, Xiaoming Liu and Fengju Shang, "Vector-based low-delay forwarding protocol for underwater wireless sensor networks," in *Multimedia Information Networking and Security (MINES), 2010 International Conference on,* 2010, pp. 178-181.

[93] Y. Bayrakdar, N. Meratnia and A. Kantarci, "A comparative view of routing protocols for underwater wireless sensor networks," in *OCEANS, 2011 IEEE-Spain,* 2011, pp. 1-5.

[94] D. Hwang and D. Kim, "DFR: Directional flooding-based routing protocol for underwater sensor networks," in *Oceans 2008,* 2008, pp. 1-7.

[95] Y. Wang, J. Gao and J. S. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking,* 2006, pp. 122-133.

[96] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Algorithmic Aspects of Wireless Sensor Networks,* Springer, 2004, pp. 123-136.

[97] Z. Guo, G. Colombo, B. Wang, J. Cui, D. Maggiorini and G. P. Rossi, "Adaptive routing in underwater delay/disruption tolerant sensor networks," in *Wireless on Demand Network Systems and Services, 2008. WONS 2008. Fifth Annual Conference on,* 2008, pp. 31-39.

[98] G. Han, C. Zhang, L. Shu, N. Sun and Q. Li, "A survey on deployment algorithms in underwater acoustic sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[99] S. Alam and Z. J. Haas, "Coverage and connectivity in three-dimensional networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking*, 2006, pp. 346-357.

[100] P. Casari and M. Zorzi, "Protocol design issues in underwater acoustic networks," *Comput. Commun.*, vol. 34, pp. 2013-2025, 2011.

APPENDIX A

3D Node Deployment Strategy Code with VBF as baseline Routing Protocol

```
set opt(chan)        Channel/UnderwaterChannel
set opt(prop)        Propagation/UnderwaterPropagation
set opt(netif)       Phy/UnderwaterPhy
set opt(mac)         Mac/UnderwaterMac/BroadcastMac
set opt(ifq)         Queue/DropTail/PriQueue
set opt(ll)          LL
set opt(energy)       EnergyModel
set opt(txpower)         2.0       ;# in watts
set opt(rxpower)         0.75      ;# in watts
set opt(initialenergy)  10000      ;# in watts
set opt(idlepower)       0.008     ;# in watts
set opt(ant)             Antenna/OmniAntenna
set opt(filters)         GradientFilter ;# options can be ;#
one or more of TPP/OPP/Gear/Rmst/SourceRoute/Log/TagFilter

set opt(position_update_interval) 0.3  ;# the length of
period to update node's position
set opt(packet_size)    50             ;#50 bytes
set opt(routing_control_packet_size) 20 ;#bytes

set opt(ifqlen)      50   ;# max queue length in if
set opt(nn)          26   ;# number of nodes
set opt(x)           500  ;# X dimension of the topography
set opt(y)           500  ;# Y dimension of the topography
set opt(z)            500
set opt(seed)        11
set opt(stop)        500       ;# simulation time
set opt(prestop)      90       ;# time to prepare to stop
set opt(tr)          "Test_9b.tr"   ;# trace file
```

```
set opt(datafile)   "Test_9b.data" ;# data file
set opt(nam)        "Test_9b.nam"  ;# nam file
set opt(adhocRouting)   Vectorbasedforward
set opt(width)           400;
set opt(interval)        10.0
set opt(range)           100 ;#range of each node in meters

if { $argc > 0 } {
  set opt(seed) [lindex $argv 0]
  set opt(nn) [lindex $argv 1]
  set opt(datafile) [lindex $argv 2]
}

puts "the file name is $opt(datafile)"
puts "the sending interval is $opt(interval)"

#
===========================================================

LL set mindelay_         50us
LL set delay_            25us

Queue/DropTail/PriQueue set Prefer_Routing_Protocols    1

# unity gain, omni-directional antennas

Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0.05
Antenna/OmniAntenna set Z_ 0
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

Agent/Vectorbasedforward set hop_by_hop_ 0

Mac/UnderwaterMac set bit_rate_  1.0e4 ;#10kbps
Mac/UnderwaterMac set encoding_efficiency_ 1
Mac/UnderwaterMac/BroadcastMac set packetheader_size_ 0
;#of bytes

# Initialize the SharedMedia interface with parameters to
make
# it work like the 914MHz Lucent WaveLAN DSSS radio
interface
Phy/UnderwaterPhy set CPThresh_  10   ;#10.0
Phy/UnderwaterPhy set CSThresh_  0    ;#1.559e-11
Phy/UnderwaterPhy set RXThresh_  0    ;#3.652e-10
#Phy/WirelessPhy set Rb_ 2*1e6
Phy/UnderwaterPhy set Pt_  0.2818
Phy/UnderwaterPhy set freq_  25       ;# 25khz
Phy/UnderwaterPhy set K_ 2.0          ;# spherical
spreading
```

```
#
===================================================================
# Main Program
#
===================================================================

#
# Initialize Global Variables
#
#set sink_ 1
set ns_ [new Simulator]
set topo  [new Topography]

$topo load_cubicgrid $opt(x) $opt(y) $opt(z)
#$ns_ use-newtrace
set tracefd    [open $opt(tr) w]
$ns_ trace-all $tracefd

set nf [open $opt(nam) w]
$ns_ namtrace-all-wireless $nf $opt(x) $opt(y)


set data [open $opt(datafile) a]

set total_number  [expr $opt(nn)-1]
set god_ [create-god  $opt(nn)]

$ns_ at 0.0 "$god_  set_filename $opt(datafile)"

set chan_1_ [new $opt(chan)]

global defaultRNG
$defaultRNG seed $opt(seed)

$ns_ node-config -adhocRouting $opt(adhocRouting) \
           -llType $opt(ll) \
           -macType $opt(mac) \
           -ifqType $opt(ifq) \
           -ifqLen $opt(ifqlen) \
           -antType $opt(ant) \
           -propType $opt(prop) \
           -phyType $opt(netif) \
           -agentTrace OFF \
           -routerTrace OFF \
           -macTrace ON\
           -topoInstance $topo\
           -energyModel $opt(energy)\
           -txPower $opt(txpower)\
           -rxPower $opt(rxpower)\
           -initialEnergy $opt(initialenergy)\
```

```
            -idlePower $opt(idlepower)\
            -channel $chan_1_

puts "Width=$opt(width)"
#Set the Sink node

set node_(0) [ $ns_  node 0]
$node_(0) set sinkStatus_ 1
$god_ new_node $node_(0)
$node_(0) set X_  150
$node_(0) set Y_  500
$node_(0) set Z_  150
$node_(0) set passive 1

set rt [$node_(0) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(0) [new Agent/UWSink]
$ns_ attach-agent $node_(0) $a_(0)
$a_(0) attach-vectorbasedforward $opt(width)
$a_(0) cmd set-range $opt(range)
$a_(0) cmd set-filename $opt(datafile)
$a_(0) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(1) [ $ns_  node 1]
$node_(1) set sinkStatus_ 1
$god_ new_node $node_(1)
$node_(1) set X_  100
$node_(1) set Y_  450
$node_(1) set Z_  100
$node_(1) set passive 1

set rt [$node_(1) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)
$node_(1) set max_speed $opt(maxspeed)
$node_(1) set min_speed $opt(minspeed)
$node_(1) set position_update_interval_
$opt(position_update_interval)
set a_(1) [new Agent/UWSink]
$ns_ attach-agent $node_(1) $a_(1)
$a_(1) attach-vectorbasedforward $opt(width)
$a_(1) cmd set-range $opt(range)
$a_(1) cmd set-filename $opt(datafile)
$a_(1) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(2) [ $ns_  node 2]
$node_(2) set sinkStatus_ 1
```

```
$node_(2) random-motion 1
$node_(2) set max_speed $opt(maxspeed)
$node_(2) set min_speed $opt(minspeed)
$node_(2) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(2)
$node_(2) set X_  100
$node_(2) set Y_  350
$node_(2) set Z_  100
$node_(2) set passive 1

set rt [$node_(2) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(2) [new Agent/UWSink]
$ns_ attach-agent $node_(2) $a_(2)
$a_(2) attach-vectorbasedforward $opt(width)
$a_(2) cmd set-range $opt(range)
$a_(2) cmd set-filename $opt(datafile)
$a_(2) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(3) [ $ns_  node 3]
$node_(3) set sinkStatus_ 1
$node_(3) random-motion 1

$node_(3) set max_speed $opt(maxspeed)
$node_(3) set min_speed $opt(minspeed)
$node_(3) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(3)
$node_(3) set X_  100
$node_(3) set Y_  250
$node_(3) set Z_  100
$node_(3) set passive 1

set rt [$node_(3) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(3) [new Agent/UWSink]
$ns_ attach-agent $node_(3) $a_(3)
$a_(3) attach-vectorbasedforward $opt(width)
$a_(3) cmd set-range $opt(range)
$a_(3) cmd set-filename $opt(datafile)
$a_(3) cmd set-packetsize $opt(packet_size) ;# # of bytes
```

```
set node_(4) [ $ns_  node 4]
$node_(4) set sinkStatus_ 1
$node_(4) random-motion 1

$node_(4) set max_speed $opt(maxspeed)
$node_(4) set min_speed $opt(minspeed)
$node_(4) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(4)
$node_(4) set X_  100
$node_(4) set Y_  150
$node_(4) set Z_  100
$node_(4) set passive 1

set rt [$node_(4) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(4) [new Agent/UWSink]
$ns_ attach-agent $node_(4) $a_(4)
$a_(4) attach-vectorbasedforward $opt(width)
$a_(4) cmd set-range $opt(range)
$a_(4) cmd set-filename $opt(datafile)
$a_(4) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(5) [ $ns_  node 5]
$node_(5) set sinkStatus_ 1
$node_(5) random-motion 1

$node_(5) set max_speed $opt(maxspeed)
$node_(5) set min_speed $opt(minspeed)
$node_(5) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(5)
$node_(5) set X_  100
$node_(5) set Y_  50
$node_(5) set Z_  100
$node_(5) set passive 1

set rt [$node_(5) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(5) [new Agent/UWSink]
$ns_ attach-agent $node_(5) $a_(5)
$a_(5) attach-vectorbasedforward $opt(width)
$a_(5) cmd set-range $opt(range)
$a_(5) cmd set-filename $opt(datafile)
```

```
$a_(5) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(6) [ $ns_  node 6]
$node_(6) set sinkStatus_ 1
$node_(6) random-motion 1

$node_(6) set max_speed $opt(maxspeed)
$node_(6) set min_speed $opt(minspeed)
$node_(6) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(6)
$node_(6) set X_  200
$node_(6) set Y_  50
$node_(6) set Z_  100
$node_(6) set passive 1

set rt [$node_(6) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(6) [new Agent/UWSink]
$ns_ attach-agent $node_(6) $a_(6)
$a_(6) attach-vectorbasedforward $opt(width)
$a_(6) cmd set-range $opt(range)
$a_(6) cmd set-filename $opt(datafile)
$a_(6) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(7) [ $ns_  node 7]
$node_(7) set sinkStatus_ 1
$node_(7) random-motion 1

$node_(7) set max_speed $opt(maxspeed)
$node_(7) set min_speed $opt(minspeed)
$node_(7) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(7)
$node_(7) set X_  200
$node_(7) set Y_  150
$node_(7) set Z_  100
$node_(7) set passive 1

set rt [$node_(7) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(7) [new Agent/UWSink]
$ns_ attach-agent $node_(7) $a_(7)
```

```
$a_(7) attach-vectorbasedforward $opt(width)
$a_(7) cmd set-range $opt(range)
$a_(7) cmd set-filename $opt(datafile)
$a_(7) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(8) [ $ns_  node 8]
$node_(8) set sinkStatus_ 1
$node_(8) random-motion 1

$node_(8) set max_speed $opt(maxspeed)
$node_(8) set min_speed $opt(minspeed)
$node_(8) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(8)
$node_(8) set X_  200
$node_(8) set Y_  250
$node_(8) set Z_  100
$node_(8) set passive 1

set rt [$node_(8) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(8) [new Agent/UWSink]
$ns_ attach-agent $node_(8) $a_(8)
$a_(8) attach-vectorbasedforward $opt(width)
$a_(8) cmd set-range $opt(range)
$a_(8) cmd set-filename $opt(datafile)
$a_(8) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(9) [ $ns_  node 9]
$node_(9) set sinkStatus_ 1
$node_(9) random-motion 1

$node_(9) set max_speed $opt(maxspeed)
$node_(9) set min_speed $opt(minspeed)
$node_(9) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(9)
$node_(9) set X_  200
$node_(9) set Y_  350
$node_(9) set Z_  100
$node_(9) set passive 1

set rt [$node_(9) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)
```

```
set a_(9) [new Agent/UWSink]
$ns_ attach-agent $node_(9) $a_(9)
$a_(9) attach-vectorbasedforward $opt(width)
$a_(9) cmd set-range $opt(range)
$a_(9) cmd set-filename $opt(datafile)
$a_(9) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(10) [ $ns_  node 10]
$node_(10) set sinkStatus_ 1
$node_(10) random-motion 1

$node_(10) set max_speed $opt(maxspeed)
$node_(10) set min_speed $opt(minspeed)
$node_(10) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(10)
$node_(10) set X_  200
$node_(10) set Y_  450
$node_(10) set Z_  100
$node_(10) set passive 1

set rt [$node_(10) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(10) [new Agent/UWSink]
$ns_ attach-agent $node_(10) $a_(10)
$a_(10) attach-vectorbasedforward $opt(width)
$a_(10) cmd set-range $opt(range)
$a_(10) cmd set-filename $opt(datafile)
$a_(10) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(11) [ $ns_  node 11]
$node_(11) set sinkStatus_ 1
$node_(11) random-motion 1

$node_(11) set max_speed $opt(maxspeed)
$node_(11) set min_speed $opt(minspeed)
$node_(11) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(11)
$node_(11) set X_  100
$node_(11) set Y_  450
$node_(11) set Z_  200
$node_(11) set passive 1
```

```
set rt [$node_(11) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(11) [new Agent/UWSink]
$ns_ attach-agent $node_(11) $a_(11)
$a_(11) attach-vectorbasedforward $opt(width)
$a_(11) cmd set-range $opt(range)
#$a_(11) cmd set-target-x -0
#$a_(11) cmd set-target-y -20
#$a_(11) cmd set-target-z -0
$a_(11) cmd set-filename $opt(datafile)
$a_(11) cmd set-packetsize $opt(packet_size) ;# # of bytes
$node_(11) move


set node_(12) [ $ns_  node 12]
$node_(12) set sinkStatus_ 1
$node_(12) random-motion 1

$node_(12) set max_speed $opt(maxspeed)
$node_(12) set min_speed $opt(minspeed)
$node_(12) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(12)
$node_(12) set X_  100
$node_(12) set Y_  350
$node_(12) set Z_  200
$node_(12) set passive 1

set rt [$node_(12) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(12) [new Agent/UWSink]
$ns_ attach-agent $node_(12) $a_(12)
$a_(12) attach-vectorbasedforward $opt(width)
$a_(12) cmd set-range $opt(range)
$a_(12) cmd set-filename $opt(datafile)
$a_(12) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(13) [ $ns_  node 13]
$node_(13) set sinkStatus_ 1
$node_(13) random-motion 1

$node_(13) set max_speed $opt(maxspeed)
$node_(13) set min_speed $opt(minspeed)
$node_(13) set position_update_interval_
$opt(position_update_interval)
```

```
$god_ new_node $node_(13)
$node_(13) set X_  100
$node_(13) set Y_  250
$node_(13) set Z_  200
$node_(13) set passive 1

set rt [$node_(11) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(13) [new Agent/UWSink]
$ns_ attach-agent $node_(13) $a_(13)
$a_(13) attach-vectorbasedforward $opt(width)
$a_(13) cmd set-range $opt(range)
$a_(13) cmd set-filename $opt(datafile)
$a_(13) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(14) [ $ns_  node 14]
$node_(14) set sinkStatus_ 1
$node_(14) random-motion 1

$node_(14) set max_speed $opt(maxspeed)
$node_(14) set min_speed $opt(minspeed)
$node_(14) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(14)
$node_(14) set X_  100
$node_(14) set Y_  150
$node_(14) set Z_  200
$node_(14) set passive 1

set rt [$node_(14) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(14) [new Agent/UWSink]
$ns_ attach-agent $node_(14) $a_(14)
$a_(14) attach-vectorbasedforward $opt(width)
$a_(14) cmd set-range $opt(range)
$a_(14) cmd set-filename $opt(datafile)
$a_(14) cmd set-packetsize $opt(packet_size) ;# # of bytes

set node_(15) [ $ns_  node 15]
$node_(15) set sinkStatus_ 1
$node_(15) random-motion 1

$node_(15) set max_speed $opt(maxspeed)
$node_(15) set min_speed $opt(minspeed)
```

```
$node_(15) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(15)
$node_(15) set X_  100
$node_(15) set Y_  50
$node_(15) set Z_  200
$node_(15) set passive 1

set rt [$node_(15) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(15) [new Agent/UWSink]
$ns_ attach-agent $node_(15) $a_(15)
$a_(15) attach-vectorbasedforward $opt(width)
$a_(15) cmd set-range $opt(range)
$a_(15) cmd set-filename $opt(datafile)
$a_(15) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(16) [ $ns_  node 16]
$node_(16) set sinkStatus_ 1
$node_(16) random-motion 1

$node_(16) set max_speed $opt(maxspeed)
$node_(16) set min_speed $opt(minspeed)
$node_(16) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(16)
$node_(16) set X_  200
$node_(16) set Y_  50
$node_(16) set Z_  200
$node_(16) set passive 1

set rt [$node_(16) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(16) [new Agent/UWSink]
$ns_ attach-agent $node_(16) $a_(16)
$a_(16) attach-vectorbasedforward $opt(width)
$a_(16) cmd set-range $opt(range)
$a_(16) cmd set-filename $opt(datafile)
$a_(16) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(17) [ $ns_  node 17]
$node_(17) set sinkStatus_ 1
$node_(17) random-motion 1
```

```
$node_(17) set max_speed $opt(maxspeed)
$node_(17) set min_speed $opt(minspeed)
$node_(17) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(17)
$node_(17) set X_  200
$node_(17) set Y_  150
$node_(17) set Z_  200
$node_(17) set passive 1

set rt [$node_(17) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(17) [new Agent/UWSink]
$ns_ attach-agent $node_(17) $a_(17)
$a_(17) attach-vectorbasedforward $opt(width)
$a_(17) cmd set-range $opt(range)
$a_(17) cmd set-filename $opt(datafile)
$a_(17) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(18) [ $ns_  node 18]
$node_(18) set sinkStatus_ 1
$node_(18) random-motion 1

$node_(18) set max_speed $opt(maxspeed)
$node_(18) set min_speed $opt(minspeed)
$node_(18) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(18)
$node_(18) set X_  200
$node_(18) set Y_  250
$node_(18) set Z_  200
$node_(18) set passive 1

set rt [$node_(18) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(18) [new Agent/UWSink]
$ns_ attach-agent $node_(18) $a_(18)
$a_(18) attach-vectorbasedforward $opt(width)
$a_(18) cmd set-range $opt(range)
$a_(18) cmd set-filename $opt(datafile)
$a_(18) cmd set-packetsize $opt(packet_size) ;# # of bytes
```

```
set node_(19) [ $ns_  node 19]
$node_(19) set sinkStatus_ 1
$node_(19) random-motion 1

$node_(19) set max_speed $opt(maxspeed)
$node_(19) set min_speed $opt(minspeed)
$node_(19) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(19)
$node_(19) set X_  200
$node_(19) set Y_  350
$node_(19) set Z_  200
$node_(19) set passive 1

set rt [$node_(19) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(19) [new Agent/UWSink]
$ns_ attach-agent $node_(19) $a_(19)
$a_(19) attach-vectorbasedforward $opt(width)
$a_(19) cmd set-range $opt(range)
$a_(19) cmd set-filename $opt(datafile)
$a_(19) cmd set-packetsize $opt(packet_size) ;# # of bytes
$node_(19) move


set node_(20) [ $ns_  node 20]
$node_(20) set sinkStatus_ 1
$node_(20) random-motion 1

$node_(20) set max_speed $opt(maxspeed)
$node_(20) set min_speed $opt(minspeed)
$node_(20) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(20)
$node_(20) set X_  200
$node_(20) set Y_  450
$node_(20) set Z_  200
$node_(20) set passive 1

set rt [$node_(20) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(20) [new Agent/UWSink]
$ns_ attach-agent $node_(20) $a_(20)
$a_(20) attach-vectorbasedforward $opt(width)
$a_(20) cmd set-range $opt(range)
```

```
$a_(20) cmd set-filename $opt(datafile)
$a_(20) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(21) [ $ns_  node 21]
$node_(21) set sinkStatus_ 1
$node_(21) random-motion 1

$node_(21) set max_speed $opt(maxspeed)
$node_(21) set min_speed $opt(minspeed)
$node_(21) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(21)
$node_(21) set X_  150
$node_(21) set Y_  400
$node_(21) set Z_  150
$node_(21) set passive 1

set rt [$node_(21) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(21) [new Agent/UWSink]
$ns_ attach-agent $node_(21) $a_(21)
$a_(21) attach-vectorbasedforward $opt(width)
$a_(21) cmd set-range $opt(range)
$a_(21) cmd set-filename $opt(datafile)
$a_(21) cmd set-packetsize $opt(packet_size) ;# # of bytes



set node_(22) [ $ns_  node 22]
$node_(22) set sinkStatus_ 1
$node_(22) random-motion 1

$node_(22) set max_speed $opt(maxspeed)
$node_(22) set min_speed $opt(minspeed)
$node_(22) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(22)
$node_(22) set X_  150
$node_(22) set Y_  300
$node_(22) set Z_  150
$node_(22) set passive 1

set rt [$node_(22) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)
```

```
set a_(22) [new Agent/UWSink]
$ns_ attach-agent $node_(22) $a_(22)
$a_(22) attach-vectorbasedforward $opt(width)
$a_(22) cmd set-range $opt(range)
$a_(22) cmd set-filename $opt(datafile)
$a_(22) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(23) [ $ns_  node 23]
$node_(23) set sinkStatus_ 1
$node_(23) random-motion 1

$node_(23) set max_speed $opt(maxspeed)
$node_(23) set min_speed $opt(minspeed)
$node_(23) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(23)
$node_(23) set X_  150
$node_(23) set Y_  200
$node_(23) set Z_  150
$node_(23) set passive 1

set rt [$node_(23) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(23) [new Agent/UWSink]
$ns_ attach-agent $node_(23) $a_(23)
$a_(23) attach-vectorbasedforward $opt(width)
$a_(23) cmd set-range $opt(range)
$a_(23) cmd set-filename $opt(datafile)
$a_(23) cmd set-packetsize $opt(packet_size) ;# # of bytes


set node_(24) [ $ns_  node 24]
$node_(24) set sinkStatus_ 1
$node_(24) random-motion 1

$node_(24) set max_speed $opt(maxspeed)
$node_(24) set min_speed $opt(minspeed)
$node_(24) set position_update_interval_
$opt(position_update_interval)

$god_ new_node $node_(24)
$node_(24) set X_  150
$node_(24) set Y_  100
$node_(24) set Z_  150
$node_(24) set passive 1

set rt [$node_(24) set ragent_]
```

```
$rt set control_packet_size
$opt(routing_control_packet_size)

set a_(24) [new Agent/UWSink]
$ns_ attach-agent $node_(24) $a_(24)
$a_(24) attach-vectorbasedforward $opt(width)
$a_(24) cmd set-range $opt(range)
$a_(24) cmd set-filename $opt(datafile)
$a_(24) cmd set-packetsize $opt(packet_size) ;# # of bytes


#Set the source node
set node_($total_number) [$ns_  node $total_number]
$god_ new_node $node_($total_number)

$node_($total_number) set  sinkStatus_ 1

$node_($total_number) set X_  150
$node_($total_number) set Y_  0
$node_($total_number) set Z_  150
$node_($total_number) set-cx  0
$node_($total_number) set-cy  0
$node_($total_number) set-cz  0
set rt [$node_($total_number) set ragent_]
$rt set control_packet_size
$opt(routing_control_packet_size)


set a_($total_number) [new Agent/UWSink]
$ns_ attach-agent $node_($total_number) $a_($total_number)
$a_($total_number) attach-vectorbasedforward $opt(width)
$a_($total_number) cmd set-range $opt(range)
$a_($total_number) cmd set-target-x 150
$a_($total_number) cmd set-target-y 500
$a_($total_number) cmd set-target-z 150
$a_($total_number) cmd set-filename $opt(datafile)
$a_($total_number) cmd set-packetsize $opt(packet_size) ;#
# of bytes
$a_($total_number) set data_rate_ [expr 1.0/$opt(interval)]

# make nam workable
set node_size 10
for {set k 0} { $k<$opt(nn)} {incr k} {
$ns_ initial_node_pos $node_($k) $node_size
}


set opt(stop2) [expr $opt(stop)+200]

puts "Node $total_number is sending first!!"
$ns_ at 1.33 "$a_($total_number) cbr-start"
```

```
$ns_ at $opt(stop).001 "$a_($total_number) terminate"

$ns_ at $opt(stop2).002 "$a_(0) terminate"

$ns_ at $opt(stop2).003  "$god_ compute_energy"
$ns_ at $opt(stop2).004  "$ns_ nam-end-wireless $opt(stop)"
$ns_ at $opt(stop2).005 "puts \"NS EXISTING...\"; $ns_
halt"


puts $data  "New simulation...."
puts $data "nodes  = $opt(nn), maxspeed = $opt(maxspeed),
minspeed = $opt(minspeed), random_seed = $opt(seed),
sending_interval_=$opt(interval), width=$opt(width)"
puts $data "x= $opt(x) y= $opt(y) z= $opt(z)"
close $data
puts "starting Simulation..."
$ns_ run
```