Islamic University of Technology

UNDER WATER DELAY TOLERANT NETWORK BY ACCOUSTIC COMMUNICATION AND DATA ANALYSIS

A Thesis Presented to

The Academic Faculty

By

Md Mominul Islam, Student ID: 122468

Khandaker Foysal Haque, Student ID: 122473

Salman Imtiaz, Student ID: 122442

In Partial Fulfillment

Of the Requirement for the Degree

Bachelor of Science in Electrical and Electronic Engineering

Academic Year: 2015-2016

A Dissertation on

UNDER WATER DELAY TOLERANT NETWORK BY ACCOUSTIC COMMUNICATION AND DATA ANALYSIS

Submitted By

Md Mominul Islam

Khandaker Foysal haque

Salman Imtiaz

Approved By

Dr. Md Ashraful Haque
Head of the Department & Professor
Department of EEE, IUT

Dr. Khandokar Habibul Kabir
Thesis Supervisor & Associate Professor
Department of EEE, IUT

Abstract

Underwater wireless sensor Network (UWSNs) are finding different application offshore exploration and ocean monitoring. In Most of these applications, the network consists of significant number of sensor nodes deployed at different depths throughout the area of interest. The sensor nodes located at the sea bed cannot communicate directly with the nodes near the surface level. They require multi-hop communication assisted by appropriate routing scheme. However this appropriateness depends not only on network resources and application requirements but also on environmental constraints. All these factors provide a platform where a resource-aware routing strategy plays a vital role to fulfill the different application requirements with dynamic environmental conditions. Realizing the fact, significant attention has been given to construct a reliable scheme, and many routing protocols have been proposed in order to provide an efficient route discovery between the sources and the sink. In this paper, we present a review and comparison of different algorithms, proposed recently in order to fulfill this requirement. The main purpose of this study is to address the issues like data forwarding, deployment and localization in UWSNs under different conditions. Later on, all of these are classified into different groups according to their characteristics and functionalities Delay-tolerant networking (DTN) is a term invented to describe and encompass all types of long-delay, disconnected, disrupted or intermittently-connected networks, where mobility and outages or scheduled contacts may be experienced. 'DTN' is also used to refer to the Bundle Protocol, which has been proposed as the one unifying solution for disparate DTN networking scenarios, after originally being designed solely for use in deep space for the 'Interplanetary Internet.' We have evaluated the network to be used in underwater data extraction purposes. Underwater terrain is very different from the terrestrial terrain, as it poses more amounts of obstructions where the normal protocols of networking tend to fail. DTN addresses this very problem through the hop-by-hop networking technique to extract data from deep sea and transport the data

to the onshore sites for further analysis. This paper has been aimed to provide the best possible solution model that can be designed using DTN to extract data from challenged underwater terrain.

Preface

The undergraduate thesis, "**UNDER WATER DELAY TOLERANT NETWORK BY ACCOUSTIC COMMUNICATION AND DATA ANALYSIS''** has been written for the completion of Bachelor of Science degree at Islamic University of Technology, Bangladesh. This thesis work and writing has been done during the year 2016 under the supervision of Dr. Khandokar Habibul Kabir, Asst. Professor of the department of Electrical and Electronic Engineering.

We would like to dedicate this thesis to our supervisor Dr. Khandokar Habibul Kabir. Without the dedicated help of him we would not be able to complete this work. We are also grateful to all of our well-wishers, who provided their perpetual support towards accomplishing this task successfully.

Table of Contents

CHAPTER

1

Introduction

1.1 Introduction

The ocean is vast as it covers around 140 million square miles; more than 70% of the Earth's surface, and half of the world's population is found within the 100 km of the coastal areas. Not only has it been a major source of nourishment production, but with time it is taking a vital role for transportation, presence of natural resources, defense and adventurous purposes. Even with all its importance to humanity, surprisingly we know very little about the Earth's water bodies. Only less than 10% of the whole ocean volume has been investigated, while a large area still remains unexplored. With the increasing role of ocean in human life, discovering these largely unexplored areas has gained more importance during the last decades. On one side, traditional approaches used for underwater monitoring missions have several drawbacks and on the other side, these inhospitable environments are not feasible for human presence as unpredictable underwater activities, high water pressure and vast areas are major reasons for un-manned exploration. Due to these reasons, Underwater Wireless Sensor Networks (UWSNs) are attracting the interest of many researchers lately, especially those working on terrestrial sensor networks.

Sensor networks used for underwater communications are different in many aspects from traditional wired or even terrestrial sensor networks (Akyildiz et al., 2005; Heidemann et al., 2006). Firstly, energy consumptions are different because some important applications require large amount of data, but very infrequently. Secondly, these networks usually work on a common task instead of representing independent users. The ultimate goal is to maximize the throughput rather than fairness among the nodes. Thirdly, for these networks, there is an important relationship between the link distance, number of hops and reliability. For energy concerns, packets over multiple short hops are preferred instead of long links, as multi-hop data deliveries have been proven more energy efficient for underwater networks than the single hop (Jiang, 2008). At the same time, it is observed that packet routing over more number of hops ultimately degrades the end-to-end reliability function especially for the harsh underwater environment. Finally, most of the time, such networks are deployed by a single organization with economical hardware, so strict interoperability with the existing standards is not required. Due to these reasons, UWSNs provide a platform that supports to review the existing structure of traditional communication protocols. The current research in UWSNs aims to meet the above criterion by introducing new design concepts, developing or improving existing protocols and building new applications.

When considering underwater sensor networks, due consideration must be given to the possible challenges that may be encountered in the subsurface environment. Continuous node movement and 3d topology are major issues posed by the host conditions. Further, some of the underwater applications, including detection or rescue missions, tend to be ad hoc in nature, some requiring not only network deployment in short times, but also without any proper planning. In such circumstances, the routing protocols should be able to determine the node locations without any prior knowledge of the network. Not only this, the network also should be capable of reconfiguring itself with dynamic conditions in order to provide an efficient communication environment. Moreover, a significant issue in selecting a system is establishing a relation between the communication range and data rate with the specific conditions. A system designed for deep water may not be suitable for shallow water or even when configured for higher data rates when reverberation is present in the environment (Chitre et al., 2008). Manufacturer's specifications of maximum data rates mostly are only useful for establishing the upper performance bound, but in practice these are not reachable with specific conditions. Users who are well funded have resorted to purchasing multiple systems and testing them in particular environment to determine if they will meet their throughput rather than fairness among the nodes. Thirdly, for these networks, there is an important relationship between the link distance, number of hops and reliability.

1.2 Overview

Underwater networks consist of a variable number of sensors and vehicles that are deployed to perform collaborative monitoring tasks over a given area. Underwater sensor network can be used to collect data, monitor pollution, and explore the underwater environment, mineral resources and aquatic life under the sea.

1.3 Network selection

We want to choose the most widely used network protocol, TCP/IP for communication in the underwater scenario. But there are some assumptions regarding the characteristics of TCP/IP:

- Existence of an end-to-end path between sender and receiver.
- Maximum round trip between any pair of nodes is not excessive.
- End-to-end packet drop probability is small.

But underwater sensor network falls under the category of 'challenged networks' characterized by extremely limited end node power, memory capacity and are prone to discontinuous connection.

Hence an alternative networking protocol- 'Delay Tolerant Network' is our approach to account for the assumptions/limitations of TCP/IP.

Characteristics of Delay Tolerant Network:

- No end-to-end connectivity required.
- Long/Variable delays can be overcome.
- Presence of storage for every router.

1.4 Research challenges

Major challenges in the design of Underwater Acoustic Networks are:

- Battery power is limited and usually batteries cannot be recharged
- The available bandwidth is severely limited
- Channel characteristics, including long and variable propagation delays, multi-path and fading problems
- High bit error rates
- Underwater sensors are prone to failures.

CHAPTER

2

Background Studies

2.1 Why acoustic?

In terrestrial sensor network, communication is basically radio based. This is due to the relatively low power needed to transmit radio messages and basically the omnidirectional nature of radio propagation. Unfortunately, the majority of the electromagnetic spectrum is significantly attenuated by seawater, making radio communication impractical in underwater networks. Optical communication might be an exception. The primary advantage of optical communication is the higher theoretical data rate due to the higher frequency signal, while the disadvantages are range and line-of-sight operation. They are also affected by scattering. Hence in underwater networks, wireless communication is typically based on acoustic links.

2.2 Formation of UW-SDTN

We apply the concept of underwater acoustic sensor network (UWASN) in collaboration with DTN to form Underwater Sensor Delay Tolerant Network (UW-SDTN). In this approach we first deploy static sensor nodes. These static nodes are anchored to the ocean bottom or sea bed and are used for data extraction. A group of static nodes surrounds a central node. This static central node is used to relay the data accumulated / collected from the static corner sensor nodes to the surface station using multi-hop paths. The multi-hop path is established by using Autonomous Underwater Vehicles (AUVs). Data is transferred from the central node to the AUVs which relay it to a boat at the surface. Transmission of data is completed when data is ultimately received at the surface sink. To minimize transmission delay, multiple AUVs and boats can be used.

2.3 Advantages of this approach

- Less energy consumed as only certain nodes (central) need to send long range signals.
- Less probability of data overflow as the corner nodes pass information to the central node, having higher storage capacity.

- Data correlation is possible as the sensor nodes are placed in small groups in a particular area.

2.4 Assumptions for the formation of S-DTN

- Instantaneous data transfer between the different components of the network
- Uniform sea floor
- Movement of AUV is not affected by water current flow or marine life
- Boat speed constant
- A 2-dimensional model to simplify a 3-dimensional real world situation

2.5 Basics of underwater Communications

Underwater acoustic communication is characterized by path loss which depends not only on the distance between transmitter and receiver but also on the signal frequency. [13] Absorption loss occur due to transfer of acoustic energy into heat .This fact implies the dependence of acoustic bandwidth on the communication distance. The absorption loss increase with increase of operating frequency and the distance between the transmitter and receiver. [13] This impose a limit on the available bandwidth within the practical constraints of finite transmission power. Consequently shorter the communication link more the bandwidth, longer the communication link less the bandwidth. And lower the Bandwidth less the data rate [5] [3]. As for example transmitting over a distance of 100km can be performed by 1 hop using a bandwidth of 1 KHz or by 10 hops using a bandwidth of 10 KHZ. Thus in exchange for a more complicated relays significant increase in information throughput can be obtained. [13]

Free space optical (FSO) waves are limited by the severe water absorption at the optical frequency band and strong backscatter from suspending particles. Its attenuation is very high and it is almost 1000 times of that of air even in the clearest water. And turbid water has more than 100 times the attenuation of the densest fog [14]. So in underwater main drawback of FSO is transmitting distance. Acoustic communication is the most versatile underwater communication technique due to its low attenuation. This is especially true in thermally stable, deep water. But in case of using acoustic waves in shallow water, performance can be adversely affected by temperature gradients, surface ambient noise, and multipath propagation due to reflection and refraction Prospects and Problems of Wireless Communication for Underwater Sensor Networks. The speed of sound in water is 4 times than that of in air [14]. But speed of the sound increases with the increase of the factors like temperature, depth, practical salinity Unit (PSU) [3]. Even though its speed in water is much slower than that of EM waves. But it is the most reliable underwater communication medium that the present technology could come up with.

2.6 Deployment and Network architecture

Underwater sensor network (UWSN) consist of variable number of sensor nodes which are deployed over a given volume to perform a collaborative monitoring. Every sensor node will perfume multi hope paths in order to reach surface sink. In compared to terrestrial sensor network UWSN is more sensitive and complicated due to its 3D nature. Akyildiz et al. (2005) [1] in his work , has proposed two communication architecture (1)two dimensional network architecture (2)three dimensional network architecture In two dimensional network architecture , A group of sensor nodes are anchored to the bottom of the ocean with deep ocean anchors. Underwater sensor nodes are interconnected to one or more underwater sinks (uw sinks) by means of wireless acoustic links. Underwater sinks are network devices in charge of relaying data from the ocean bottom network to a surface station. To achieve this objective, uw-sinks are equipped with two acoustic transceivers, namely a vertical and a horizontal transceiver. The horizontal transceiver is used by the uwsink to communicate with the sensor nodes in order to: (i) send commands and configuration data to the sensors (uwsink to sensors); (ii) collect monitored data (sensors to uwsink). The vertical link is used by the uw-sinks to relay data to a surface station. In deep water applications, vertical transceivers must be long range transceivers as the ocean can be as deep as 10 km. The surface station is equipped with an acoustic transceiver that is able to handle multiple parallel communications with the deployed uw-sinks. It is also endowed with a long range RF and/or satellite transmitter to communicate with the onshore sink (os-sink) and/or to a surface sink (s-sink). Sensors can be connected to uw-sinks via direct links or through multi-hop paths. The direct link connection is the simplest way to network sensors, but it may not be the most energy efficient solution. Because of increased acoustic interference due high transmission power direct links are likely to reduce the network throughput. In case of multi-hop paths, the data produced by a source sensor is relayed by intermediate sensors until it reaches the uw-sink. This may result in energy savings and increased network capacity though increases the complexity and routing functionality. As energy and capacity are prime factors for underwater communication, in UW-ASNs the objective is to deliver event features by exploiting multi-hop paths and minimizing the signaling overhead necessary to construct underwater paths at the same time three dimensional underwater networks are used to detect and observe phenomena that cannot be adequately observed by means of ocean bottom sensor nodes, i.e., to perform cooperative sampling of the 3D ocean environment. In three-dimensional underwater networks, sensor nodes float at different depths in order to observe a given phenomenon. One possible solution would be to attach each uw-sensor node to a surface buoy, by means of wires whose length can be regulated so as to adjust the depth of each sensor node However, although this solution allows easy and quick deployment of the sensor network, multiple floating buoys may obstruct ships navigating on the surface, or they can be easily detected and deactivated by enemies in military settings. Furthermore, floating buoys are vulnerable to weather tampering and pilfering But a slightly different idea was proposed in Pompili et al. (2006) [13]. Here author suggested to attach each uw-sensor node to a surface buoy, by means of wires whose length can be regulated to adjust the depth of each sensor node instead of sensor node floats at different depth.

2.7 Localization:

There are few application which require the time and location of the sensed data. So Localization is very important for time critical application which need timely information [15]. Localization in underwater is challenging because radio frequency is highly attenuated in underwater, thus GPS technology is not feasible there. Most localization schemes need to know the location of some nodes which are called anchor node or reference node. Such localization schemes are broadly divided into two categories 1.Range based scheme 2. Range free scheme. In UWSNs, acoustic channels are naturally employed, and range measurements become much more accurate when we use Acoustic channel instead of radio [16] [6] [17]. Thus, Due to having the characteristics like low communication bandwidth, node mobility and three dimensional node deployment in UWSNs, range based schemes have become a good choice [18] [6]. J.E Garcia, Chandrashekhar , chandrashekhar and yoo sang choo proposed few localization schemes [19] [20] [21]. These solutions are mainly designed for small-scale networks (usually with tens of nodes or even less). In case of large scale UWSNs, hundreds or thousands of sensor nodes are deployed in a wide underwater area. In Erol et al. (2007), the authors proposed an idea of Dive and Rise (DNR) for positioning, the novel idea of using DNR beacons for localization [22]. IN this method, static anchor nodes are replaced by mobile DNR beacons. The major drawback of this DNR scheme is that it requires large number of expensive DNR beacons. Chen et al (2009) improved the scheme and reduced the expense by decreasing the requirements of DNR beacons [23]. They replace the DNR beacons with four types of nodes- surface buoys, Detachable Elevator Transceivers (DETs), anchor nodes, and ordinary sensor nodes. The assumption for this schemes are 1. All the sensor nodes have pressure sensor in order to provide depth position or z-coordinate information. 2. Static Network.

2.8 Reliability:

The main challenging factor for under water networking system is to deliver the collected data packet to the surface sink than to forward the data to the control center. Transmission control protocol (TCP) and other congestion control mechanisms shows many difficulties in underwater wireless multi hop network [24] [25] [26] [11] [27]. TCP is a protocol that works based on end to end connection technique and it needs 3 way handshake between the sender and the receiver before the main data packet transmission starts. In case of UWSNs, we have to transmit only a few bytes in each packet, for such a small volume of data it becomes a problem for TCP as it follows the 3 way handshake mechanism. IN case of acoustic communication, propagation time is larger than the transmission time which leads us to bandwidth delay product problem [3] [28]. IT is considered in TCP that only congestion is responsible for packet data losses, so TCP only focuses on the congestion control mechanism that try to decrease the transmission rate But the error prone acoustic channel and the failures of the nodes can also be a reason for data packet losses in UWSNs .So To maintain throughput efficiency it is not necessary to decrease the data transmission rate. [3] For reliability TCP needs end to end ACK and retransmission strategy but it will cause poor through put and longer transmission time. IN case of UDP, it doesn't offer any flow control or congestion control mechanism rather it just drops the packet without creating any scope for recovery or retransmission which results in total loss of the data packet. [24] Larger packets incur higher loss rates where smaller packet face greater overhead. So packet size directly affects the Reliability. It is mentionable that longer packets help to increase the collisions in the network but this is preferable only when the link quality is good enough. However experiments have shown that error probability is proportional to the data packet length [24]. And in case of multi hop wireless links, end to end routes available in the network defined the link quality. Error control mechanism is a major issue in reliability. A successful transmission highly depends on the technique that we use for error control mechanism. Depending on the properties of the wireless channel packet sizes are determined. For example bad channel conditions require smaller packet size, error detection and retransmission mechanism whereas larger packets are preferable for good channel condition. Moreover channel access rates are affected by increase in packet sizes, thus the traffic on the channel is also affected. Which finally affects the number of collision and probability of successful carrier sense. Underwater condition is very transient for wireless networking. So more reliable and more adaptive protocols are needed for successful transmission of the data packets. Even though some networking protocol play a decent role in underwater wireless sensor networking which discussed later on.

CHAPTER

3

Routing Protocols

3.1 ROUTING PROTOCOLS FOR UWSN

3.1.1 Directional Flooding-Based Routing (DFR)

Dynamic conditions and high packet loss degrade the reliability which causes retransmission. In the above routing protocols which were intended to improve reliability didn't consider link quality. To improve the reliability Dongseung and Daeyoup (2012) proposed Directional flooding based routing (DFR) protocol [29]. authors suggested , DFR makes more nodes operate as the next forwarding nodes in order to transmission in DFR increase the possibility that a packet reaches the sink reliably .Otherwise, a few forwarding nodes are enough to make the packet approach the sink reliably . This forwarding activity is performed per hop. Authors like Syed and Mohammad (2012) [30] and Daeyup and Dongkyun [29] worked on this field but Daeeyup and Dongkun (2008) [29] have the most prominent research work on DFR. Authors suggested, a data packet is broadcast by source node having its location information and one parameter this parameter contains an angle known as BASE ANGLE which is set to its predefined minimum value according to network density. When a node receives a packet then calculates the angle between two vectors, from source node to itself and from itself to sink and this angle is called CURRENT ANGLE. When node receives a packet it compares Base angle with current angle and decide whether to forward a packet. If the nodes current angle is smaller than the base angle then it is considered as out of flooding scope and it is discarded. When the current angle is higher than the base angle then the receiving node adjusts the base angle to maintain the link quality with its neighbors. Two conditions must be satisfied by all the nodes to maintain the link quality (1) the current angle of the neighbors must be larger than that of current angle of the forwarder. (2)The distance from the neighboring node a sink must be less than that of the distance of the forwarder

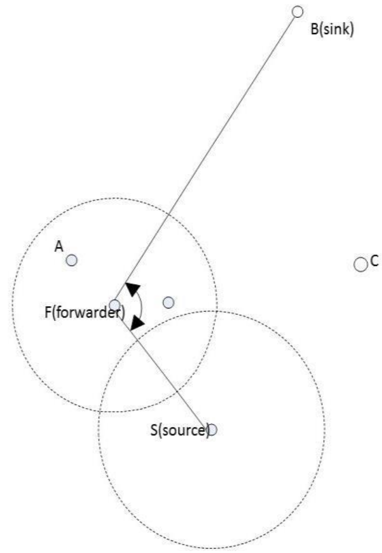

Figure 1: Directional Flooding Based Routing Protocols

to the sink. Nodes will not transmit packet until the above conditions are not fulfilled. Then forwarder floods the packet with a source location and a new base angle is set.

3.1.2. Distributed Underwater Clustering Scheme (DUCS):

Sensor nodes are powered by batteries and these are difficult to recharge or replace, so energy saving is a major concern for UWSN. The design of robust, scalable and energy-efficient routing protocols in this type of networks is a fundamental research issue. The existing multi-hop ad hoc routing protocols are not adequate because they apply a continuous exchange of overhead messages (proactive ad hoc routing) or employ a route discovery process based on the flooding technique (reactive ad hoc routing); these mechanisms are inefficient tools in large scale underwater networking because they consume excessive energy and bandwidth resources. Domingo and prior(2007) presented Distributed Underwater clustering Scheme(DUCS) which is a new distributed energy-aware routing protocol designed for long-term non-time-critical aquatic monitoring applications using UWSNs with random node mobility [31]. DUCS is an adaptive self-organizing protocol where clusters are formed using a distributed algorithm. It is supposed that underwater sensor nodes always have data to be sent to the sink and that they can use power control to adjust its transmission power. In DUCS the nodes organize themselves into local clusters and one node is selected as cluster-head for each cluster. All non-cluster head nodes transmit their data to their cluster head by single hop transmission. After cluster head nodes receive data from al the cluster members, they perform signal processing function like aggregation

And then sent the data to the sink but this time they use multi hop routing technique. Cluster heads are responsible for intra-cluster Communication and inter-cluster communication. Aided by the aggregation techniques effective non-redundant data can be extracted by the cluster head and the then data is sent to the sink, so it makes this protocol an energy efficient. Besides, to avoid fast draining of the batteries of specific underwater sensors, DUCS incorporates randomize rotation of cluster head among the sensors. The Function of operation is divided into two rounds, First round is called setup in which clusters are formed and in the second round called network operation, transfer of data is completed. During the second round, several frames are transmitted to each cluster head where every frame is composed of a series of data messages that the non-cluster head sensor nodes transmit to the cluster head using a schedule. Simulation has shown that DUCS not only achieves high packet delivery ratio, but also considerably reduces the network overhead and continues to increase throughput consequently. Though DUCS is energy efficient, it has some performance issues. The cluster structure can be affected by the movement of the nodes due to water current which may lead to decrease of the cluster life. During the second round (Network Operation), only a cluster head can transmit the collected data to another cluster head. So, if water current move the two cluster heads such far that they can't communicate directly and even a few non cluster head nodes are available between them then network will interrupt.

3.1.3 Focused beam routing (FBR)

Focused Beam Routing (FBR): Without knowing the exact location of the nodes, it becomes difficult for network to broadcast a large number of data packets. And this will eventually decrease the transmission rate. Jornet et al. (2008) proposed Focused beam routing protocol just to avoid such unnecessary flooding [32]. This is suitable for networks containing both static and mobile nodes and they are not necessarily synchronized to global clock. A source node must know its location and the location of the destination but not necessarily it has to know the location of every other nodes. FBR performs flooding for routing data packets and the flooding is restricted by the transmission power. This scheme employs various transmission power levels in order to minimize the energy consumption in underwater sensor networks. The given figure explains how a data packet is forwarded. Here, node A has the data packet and it needs to be sent to the destination node B. For this node A issue a multicast request to send RTS to its neighbor nodes. This RTS packet contains the location of source A and the destination node B. Initial transaction is performed at the lowest possible power level and power can be increased if no node is found as the next hop in the communication range provided by the power level. But Receiving node doesn't decide which power level to be used instead they considered open loop power control and power level is decided by the transmitting node.

They also considered that there are finite number of power level from P1 to PN which can be increased from one level to the upper level when needed [33]. For each power level, there is a definite transmission radius. The nodes have to be within the corresponding radius of the defined power level to receive the signal which can be detected. Now the locations of the nodes which received the multicast RTS earlier from node A is determined. Fig. 2. Illustration of FBR routing protocol: nodes within the transmitter's cone θ are candidate relays This location is relative to the line AB. Nodes which lies within the cone of angle $\pm\theta/ 2$ emanating from the source node A towards the destination node B , are considered as the eligible nodes for next hop . If a node is eligible one, this will response to the received RTS. But in real life FBR may face some difficulties. First, due to transient state of the underwater condition, and due to heavy current, it may happen that, no node is within the cone of $\pm\theta/2$. Sometimes it may happen that, some of eligible node remain outside of the forwarding cone angle. In this case it can't find the eligible node for the next hope within the forwarding zone, source node needs to rebroadcast the RTS. Thus overhead will increase which will eventually disturbs the transmission. Sink is considered to be fixed here which makes the network more restricted.

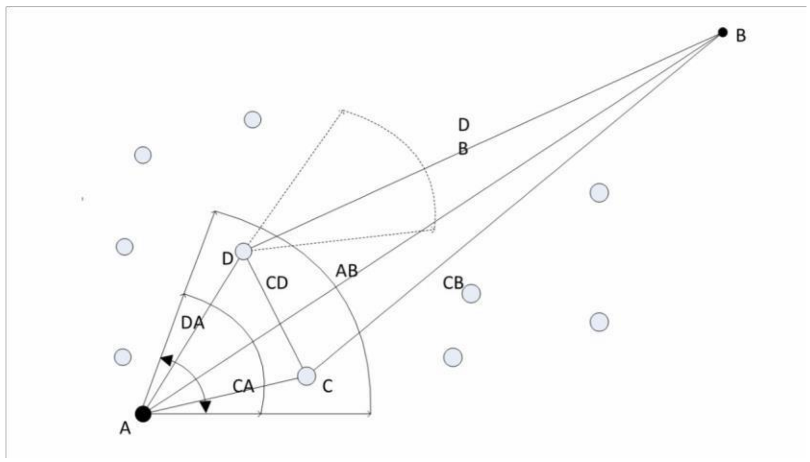

Figure 2: Focused Beam Routing Protocol

3.1.4 Vector Based Forwarding (VBF)

A location based routing approach. No state information is required on the sensor nodes and only a small fraction of the nodes are involved in routing. Data packets are forwarded along redundant and interleaved paths from source to sink, which helps handle the problem of packet losses and failures .assume that every node already knows its location and each packet carries the location of all nodes and final destination. Here the idea of a vector like a virtual routing pipe is proposed and all the packets are forwarded through this pipe from the source to the destination. Only the nodes closer to this pipe or vector from source to the destination can forward the messages. In order to increase the robustness and overcome these problems, an enhanced version of VBF called Hop-by Hop Vector-Based Forwarding (HH-VBF) has been proposed by Nicolaou et al. (2007). They use the same concept of virtual routing pipe as used by VBF, but instead of using a single pipe from source to destination, HH-VBF defines per hop virtual pipe for each forwarder. In this way, every intermediate node makes decision about the pipe direction based on its current location. By doing so, even when a small number of nodes are available in the neighborhood, HH-VBF can still find a data delivery path. VBF has some serious problems. First, the use of a virtual routing pipe from source to destination as the creation of such pipe can affect the routing efficiency of the network with different node densities. In some areas, if nodes are much sparsely deployed or become sparser due to some movements, then it is possible that very few or even no node will lie within that virtual pipe, which is responsible for the data forwarding; even it is possible that some paths may exist outside the pipe. Ultimately, this will result in small data deliveries in sparse areas. Second, VBF is very sensitive about the routing pipe radius threshold, and this threshold can affect the routing performance significantly; such feature may not be desirable in the real protocol developments. Moreover, some nodes along the routing pipe are used again and again in order to forward the data packets from concrete sources to the destination, which can exhaust their battery power. Other than these issues, VBF has much communication overhead due to its 3-way handshake nature, while during this, it does not consider the link quality.

3.1.5 Depth Based Routing (DBR)

Most of the routing protocols assume that the full dimensional location information of all sensor nodes in a network is known in prior through a localization process, but it is another challenging issue to be solved in UWSNs. Yan and shi (2008) proposed Depth based Routing protocol DBR which doesn’t require full-dimensional location information of sensor nodes rather it needs only local depth information, which can be easily obtained with an inexpensive depth sensor that can be equipped in every underwater sensor node [34] . Authors suggested that multiple data sinks placed on the water surface are used to collect the data packets from the sensor nodes. In DBR, a sensor node distributive makes its decision on packet forwarding, based on its own depth and the depth of the previous sender. This is the main idea of DBR. When a node receives a packet it first retrieve the packets previous hop which is embedded in the packet. The receiving node then compares its own depth with the depth of the previous hop. If the receiving node is closer to the water surface then it consider itself to be qualified forwarder and will forward the data. Otherwise it just drops the packet as it comes from a node which is closer to the surface. This process will be repeated until the data packet reaches any of the data sinks. Packets received at any of the data sinks is considered as the successful delivery. But DBR has some major drawbacks, DBR operates only in greedy mode which is not capable of achieve high delivery ratio in sparse areas.

3.2 Evaluation Procedure

Different types of designing philosophies and application requirements have to be considered in case of most of the routing protocols proposed for UWSNs. None of them can work efficiently for all the performance parameters like network size, localization method, reliability, node mobility etc. Due to large variations in the performance parameters it is very difficult to present a comprehensive evaluation for a large varieties of routing protocols.

Analytical modeling, real deployment, and numerical simulation are the most commonly used techniques in order to analyze the performance of terrestrial and underwater acoustic sensor. And a table will be here containing the comparisons.

Protocol	End-to-End OR hop-by-hop	Delivery Ratio	Energy Efficiency	Delay Efficiency	Localization requirement	Reliability	Performance
DFR(Daeyoup and Dongkyun, 2008)	Hop-by-hop	Medium	Medium	Medium	Needed	High	Medium
DUCS (Domingo And Prior, 2007)	Hop-by-hop	Medium	High	Low	Not Needed	Low	Low
DBR (Yan et al., 2008)	Hop-by-hop	High	Medium	High	Needed	High	High
VBF (Xie et al., 2006b)	End-to-end	Low	Medium	Low	Not Needed	Low	Low
FBR (Jornet et al., 2008)	Hop-by-hop	Medium	Medium	High	Partially Needed	Medium	High

CHAPTER

4

Our Proposed Network

4.1 Objective

The primary objective of using the network is to collect underwater data, sense and record them through the deployed sensors and transport them to sinks located on-shore. At first, it is essential to decide which part of the vast ocean is needed to be monitored. This is depended on the application for which the network is to be established at the first place.

4.2 Real world scenario

Real World Scenario describes a real world setting where we are interested to implement the network.

4.2.1 Placement of sensor nodes

In our chosen scenario, we would like to monitor ocean around few hundred meters off the coast. Thus the sensors are needed to be deployed in such a way that they form a line that is almost parallel to the coastline. Around each central sensor node, six corner nodes are placed considering a small radius, in a circular path, on the circumference of the central node with equal distances among them as depicted in fig 3(a). Each corner node will record data and transfer them to its respective central node. Number of central nodes to be placed is subject to the length of ocean to be monitored and the maximum allowable node-to-node separation.

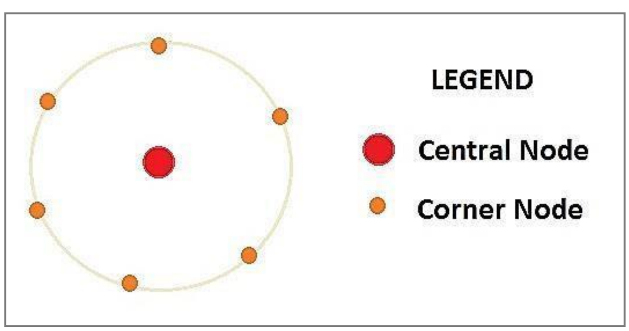

Fig. 3(a)

Knowing the depth of water at that particular distance from the coastline is of paramount importance. The seabed is not uniform and also there are waves which makes it very difficult to measure the actual depth at every point along the line. In that case, simplification of the scenario and an approximation of the depth can be very useful for the sake of advancing with the analysis.

4.2.2 Communication in the network

In underwater network, using acoustic waves is the most preferred form of communication. Since the channel is cause to experience strong signal attenuation and low data rates compared to terrestrial communication, selection of sensor ranges of all acoustic modems are vital to designing a network.

4.2.2.1 Purpose of AUV

When the ocean depth is significantly high, it is impractical to assume that every central node will forward data to the boat on the surface in a single hop. Due to the limited transmission range of sensor modems, at least one intermediate node has to be there to make up for this constraint. This is where the Autonomous Underwater Vehicle (AUV) comes in. It is basically a crewless, untethered submersible robot which operates independent of direct human control.

The AUV is programmed to move backwards and forwards over a particular path, keeping a fixed height from the sea floor. It is important to note that the AUV has to be within the range of at least one sensor node at all times as it moves along the path. Furthermore, the AUV has to have sufficient transmission range so that it can communicate with the node i.e. usually a boat above it. And if it fails to cover this distance another AUV can always be placed above the existing AUV to take care of the range gap.

4.2.2.2 Movement of boat

The boat is considered to use two forms of network – acoustic and radio waves. Acoustic to communicate with the AUV and radio waves for terrestrial communication with the sinks on-shore. Suppose there exists two on-shore sinks at point A and point B, lying on the coastline, separated by few kilometers of distance between them. We think of a boat to start from point A as it moves roughly in a parabolic path to reach the shore at point B as shown in Fig. 3(b) and Fig. 3(c), cruising above line where nodes were deployed and participating in communication with AUV and sinks whenever they lie within range.

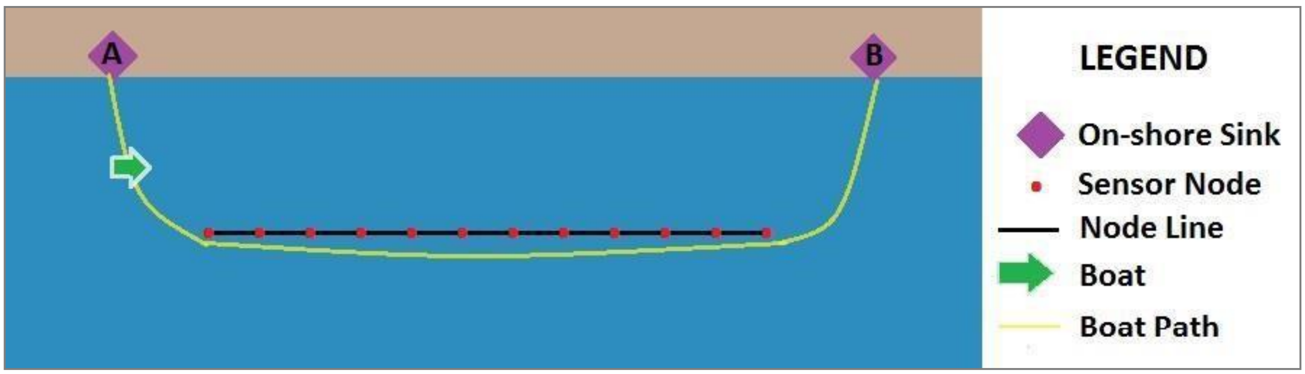

Fig. 3(b) Real world scenario, view from top

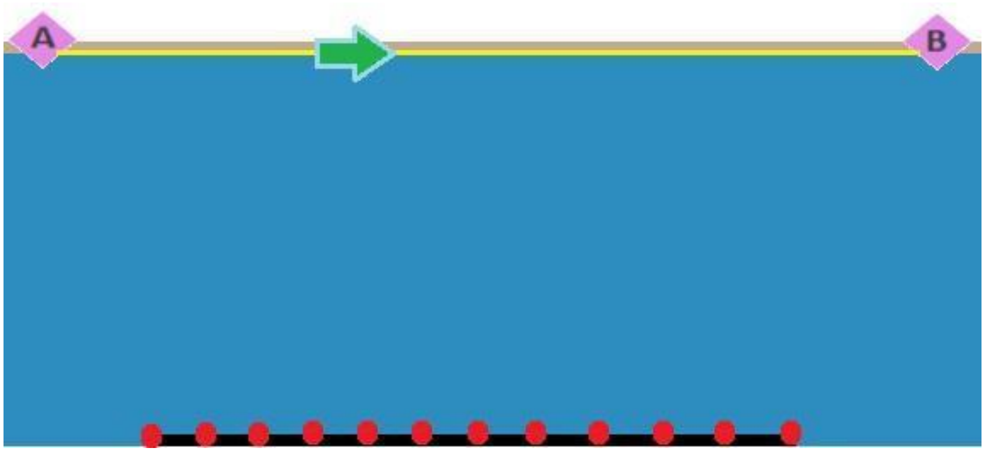

Fig. 3(c) Real world scenario, view from side

4.2.3 Application of delay tolerant network

Creating an effective network in this kind of challenged environment is quite troubling since an end-to-end connectivity cannot be assured at all times.

Sensor node generates data continuously but cannot transmit it instantaneously because an AUV is not always there within its range to receive it. Therefore sensor node needs to store the data volume in its storage until it sees an AUV. However, this kind of operation is not supported by the existing TCP/IP protocol. The necessity of a “store-and-forward” approach of data transfer is met by a new standard in communication called Delay Tolerant Network (DTN).

In simple terms, each nodes that holds data is called “custodian” of those data. The custodian keeps the data in its storage until it receives a proper acknowledgement after successfully forwarding the data to the next node. The next node now becomes the custodian, this is called “custody-transfer”. Likewise, the data moves along the network until it reaches the end point receiver, i.e. in our case the sinks located on-shore. Application of DTN makes networking simple and possible, it minimizes packet loss and improves overall efficiency of the network.

CHAPTER

5

Building the Simulator

5.1 Background

To test for performance and effectiveness of our proposed network model, it is needed to be implemented in real world and only through vigorous experimentation and subsequent result analysis, the rationality of the network can be judged. Unfortunately, due to limitations of time and money it was not possible for us to carry out the experiments materially. But there is a plausible alternative approach, which is to create a much simplified simulated environment of the real world scenario where the network is chosen to be implemented. This inspired us to build an underwater network simulator from scratch, particularly for our real world scenario.

5.2 Simulator developed in Net logo

For realistic imitation, the network model has to be three-dimensional i.e. 3D to account for distances in all three directions. But for simplification of modeling and through consideration of few assumptions, it seemed reasonable to do the modeling in two-dimension i.e. 2D.To develop our simulator, we have used *Net logo.* It is basically an agent-based programming language and integrated modeling environment. *Net Logo* is a free and open source software, under GPL license. It is written in Scala and Java and runs on the Java Virtual Machine. Net logo can be downloaded from the website http://ccl.northwestern.edu/netlogo

The figure in the following page Fig. 4(a) shows a sample simulator world view, formed after configuring different variables during setup procedure.

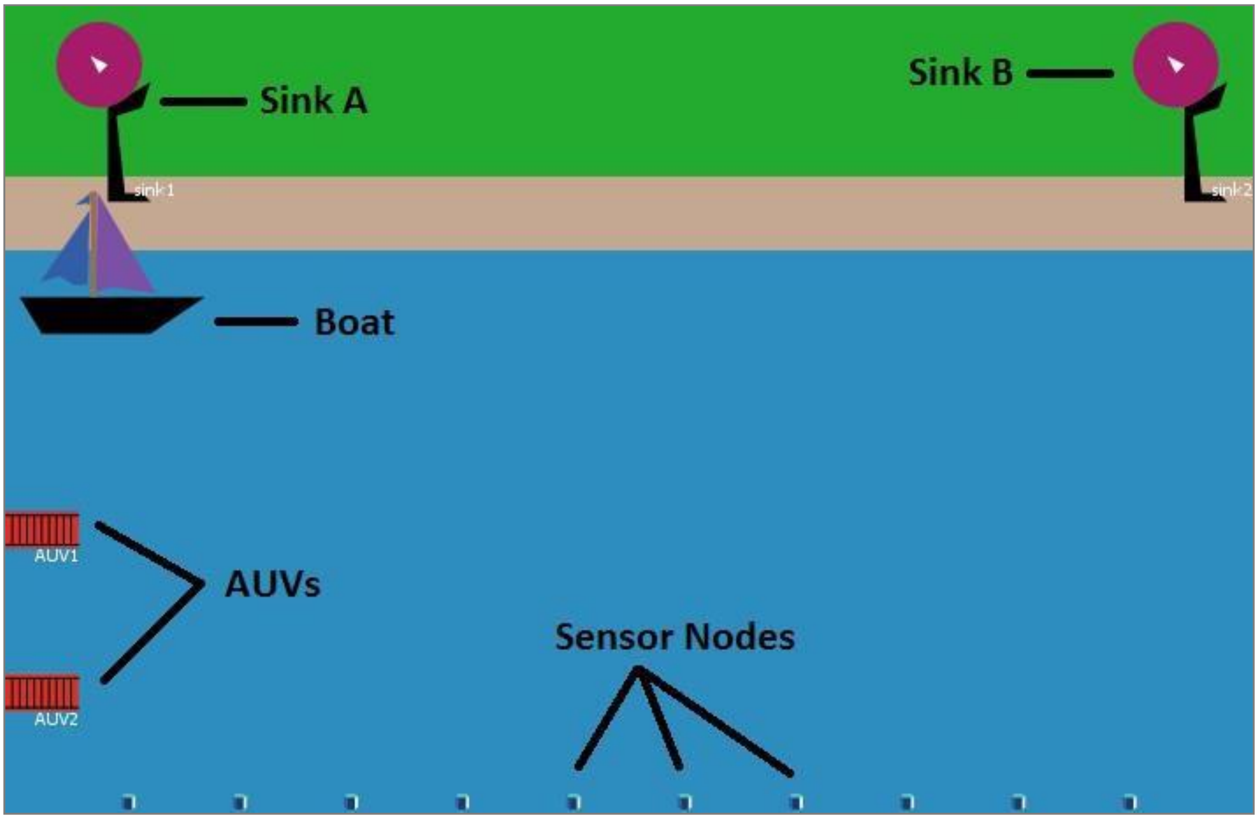

Fig. 4(a) World view in simulator

5.2.1 The Interface

Another important part of the simulator is its interface which is like a control panel where all different sliders, chooser, buttons and switches exist. Fig. 4(b) illustrates the graphical user interface (GUI) and part of the interface consists of monitors and plot viewer as depicted in Fig. 4(c).

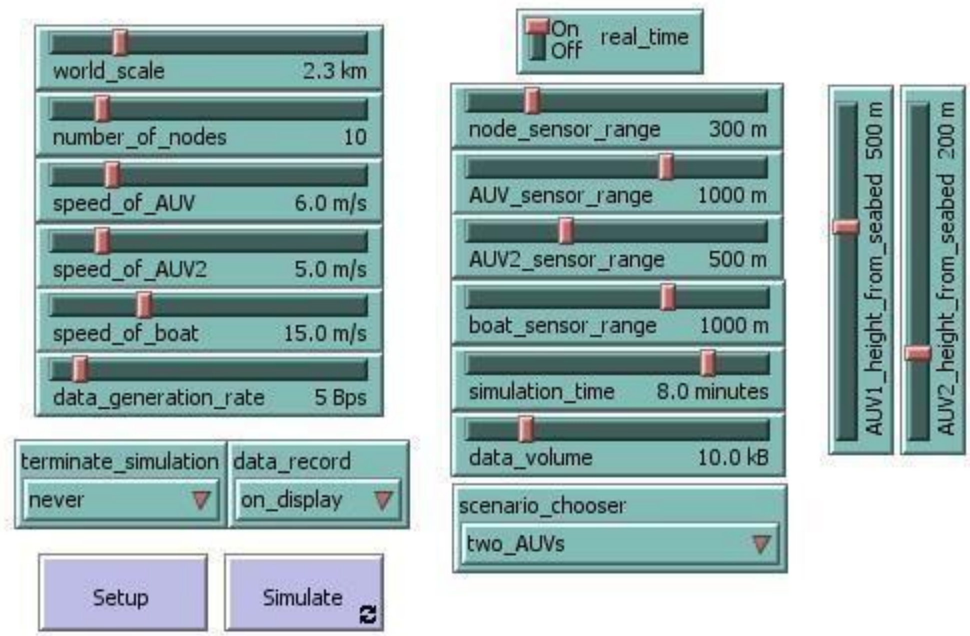

Fig. 4(b) Graphical User Interface – Setup and Control Panel

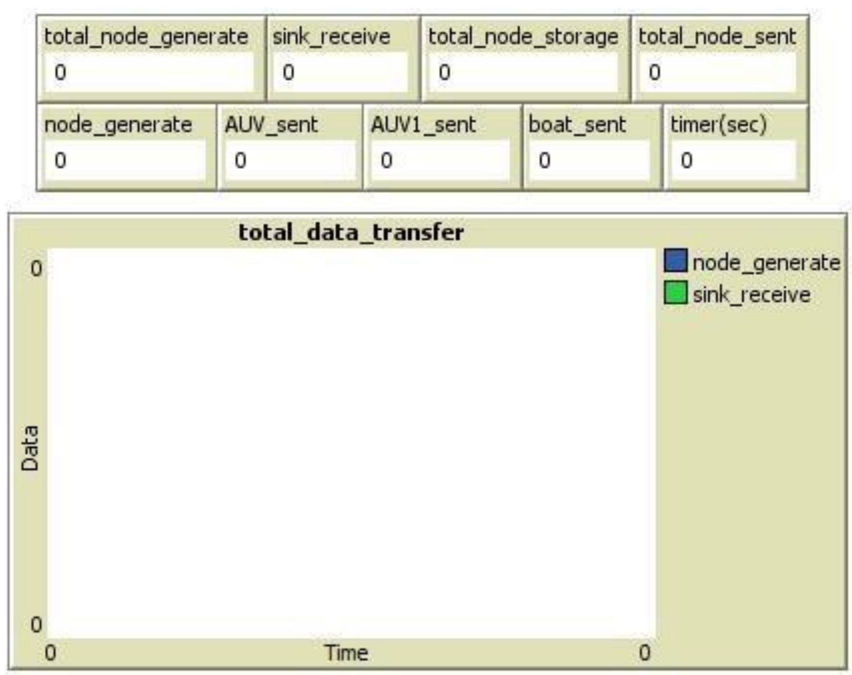

Fig. 4(c) Graphical User Interface – Monitors and Plot Viewer

5.2.1.1 Sliders

Net logo defines sliders as global variables, which are accessible by all agents. They are used in models as a quick way to change a variable without having to recode the procedure every time. Instead, the user moves the slider to a value and observes what happens in the model.

In our simulator, we have used sliders to control the following variables:

World Scale – The Net logo world is made up of cells called patches. Each patches have its own identity defined by its co-ordinates. In our model, there are a total of 1683 patches, 51 patches along x-axis and 33 patches in the y-axis, giving the world frame a rectangular shape. The horizontal axis is scaled by a scaling factor, calculated by using the value (in km) set by this slider, so that each patch defines real world length in meters.

scaling_factor = (world-width / (world_scale * 1000))

World-width means the total number of patches that exist in the horizontal axis. In our case, it is 51.The vertical axis is scaled by using the same scaling factor.

Number of Nodes–The user can place any number of sensor nodes by sliding the number_of_nodes slider before setting up the simulator. Number of sensor nodes has to be even numbered. The in-built algorithm evenly distributes them in the seabed space. The spacing between every adjacent node is same. It is to be noted that, only central nodes are considered by the simulator, in fact there exists six additional corner nodes surrounding every central node.

Speed of AUV 1– The user has the opportunity to set speed_of_AUV of his/her choice through use of this slider. The speed is calibrated in meters per second using the same scaling factor used to define world dimension. AUV 1 lies above AUV 2, so naturally AUV 1 is considered to have greater speed than AUV 2. Generally, speeds of AUV are much lower than other watercrafts like boats, submarines etc.

Speed of AUV 2 - The user has the opportunity to set a speed of his/her choice using speed_of_AUV2 slider. Again, the speed is calibrated in meters per second using the same scaling factor used to define world dimension. AUV 2 lies below AUV 1, so naturally AUV 1 is considered to have greater speed than AUV 2. It is AUV 2 that communicates directly with the nodes and perform 'data mulling'.

Speed of Boat–The speed of boat is to be defined by the user. Again, the speed is calibrated in meters per second using the same scaling factor used to define world dimension. There can be more than one boat in the scenario, in that case, all boats are assumed to cruise at the same speed as set by speed_of_boat slider.

Data Generation Rate–Sensors in node are programmed to sense and record numerous underwater data. We assumed these generated data by each node as chunk of bytes that are created periodically each second. By the data_generation_rate slider user can arbitrarily set this rate in unit of bytes per second.

Node Sensor Range - All underwater devices and vehicles use acoustic waves for communication. Acoustic waves have limited reach well defined by the range parameter of the acoustic modem used in the system. If any one of the two systems move away while communicating, the link between them would be disrupted as their separation distance exceeds the reach of the low-range device. Each node creates link with AUV 2 as the vehicle becomes available within the range of node which is defined by node_sensor_range parameter. Node modem range is assumed to be lower than AUV 2's range. Although the user has absolute freedom to select any range of his/her choice for this parameter. Data transfer occurs only when node and AUV 2 have an established link between them. The simulator represent links by a thick yellow lines as shown in Fig. 4(d).

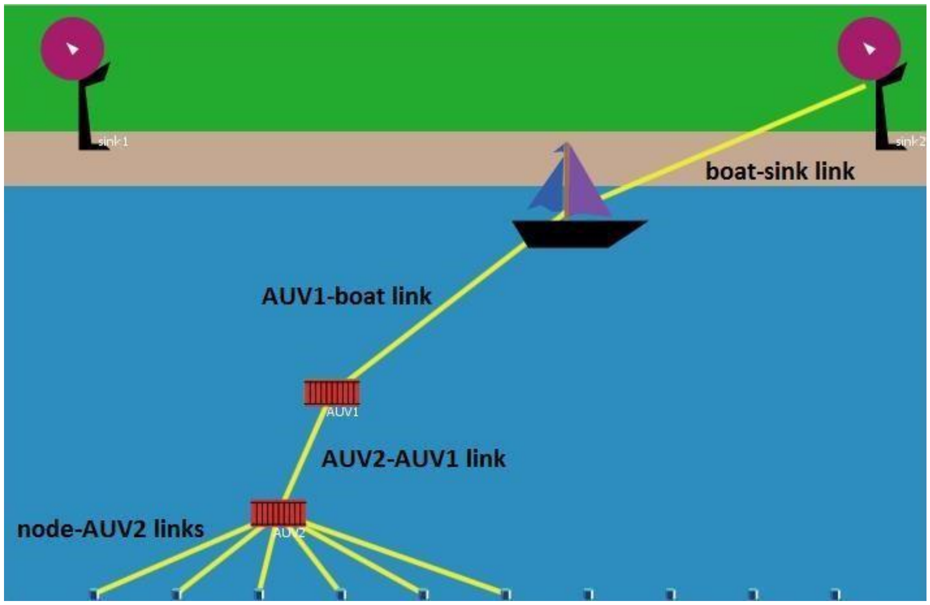

Fig. 4(d) Existing links are presented by thick yellow lines

AUV 2 Sensor Range – AUV 2 connects with both sensor nodes and AUV 1. Data bytes generated by the nodes are stored in each node's storage until AUV 2 is within reach, hence data is forwarded to AUV 2's storage. Usually, multiple node-AUV 2 links are existent at all times. These links are continuously being made and broken with the movement of AUV 2. By default AUV 2 is set to have higher range than sensor nodes. Thus these links' existence depends upon the each node's transmission capability. On the other hand, AUV 2 itself forwards its stored data to the next hop, in this case AUV 1, whenever AUV 2-AUV 1 link is established. This link's existence depends upon AUV 2 sensor range as set by the user using AUV2_sensor_range slider.

AUV Sensor Range - AUV 1 communicates with both boat and AUV 2. Data bytes collected by AUV 2 from the nodes are stored in AUV 2's storage until AUV 1 is within its reach, hence data is forwarded to AUV 1's storage. This way AUV 1 becomes the custodian of the successfully received data. By default AUV 1 is set to have higher range than AUV 2. Thus this link's existence depends on AUV 2's transmission capability. On the other hand, AUV 1 itself forwards its stored data to the next hop i.e. the boat whenever boat-AUV1 link is established. This link's existence depends upon AUV 1 sensor range as set by the user using AUV_sensor_range slider.

Boat Sensor Range - As mentioned earlier, boat uses two forms of communication using both acoustic waves in underwater communication with

AUV 1 and radio waves in Wi-Fi terrestrial communication with sinks on-shore. Data bytes stored in AUV 1 storage is received by boat whenever boat is available within AUV 1's reach. The quality of terrestrial communication is much better in comparison with underwater network. The user specified boat_sensor_range parameter actually defines the Wi-Fi range of boat-sink links made in the air.

AUV 1 Height from Seabed – The position of AUVs are imperative to designing an underwater network. Proper placement of AUV is made through exploitation of the ocean's depth. This would maximize the utilization of the discontinuous links that are created. Users can define AUV1_height_from_seabed value in meters using this slider.

AUV 2 Height from Seabed - Users can define AUV2_height_from_seabed value in meters using this slider. Again, the placement of AUVs are imperative to designing an underwater network. Distance of AUV from sea floor along with sensor ranges governs the duration of survival of link between two bodies. By identifying depth of ocean, one can easily calculate the distance of AUV 2 from water surface. Almost all AUVs in practice have limited depth of operation, this reason accentuates the importance of setting this parameter sensibly.

Simulation Time – Once any simulation is run, it is needed to be terminated either manually or automatically. A running simulation can terminate spontaneously when situation arise as set by a predetermined condition. The simplest way to do this is to affix a duration of time for which simulation will run and stop when timer expires. User can set simulation_time slider in minutes to do the above.

Data Volume – There is another approach for terminating a running simulation. For instance, if the user wants to see how much time it takes to transfer a known volume of data, he/she can easily do that by setting a fixed amount of data with the help of data_volume slider for reception at the sink. When sink_receive parameter reaches data_volume value, simulation expires instantaneously irrespective of amount of data actually been generated by the nodes at that particular time.

5.2.1.2 Switches

In Net logo, switches are a visual representation for a true/false global variable. User may set the variable to either on (true) or off (false) by flipping the switch. In our simulator, we have used only one switch:

Real Time Switch–The simulator has been developed by keeping the ease of use and flexibility in mind. While setting up the simulator, users can easily implement determined values for variables before actually running the simulation. But provision is there to manipulate several different parameters while simulation is already on the run. Through the use of real_time switch, i.e. by setting it to 'On' one may vary sensor ranges and speeds of AUVs, boat etc. But during normal operation the switch is set to 'Off'.

5.2.1.3 Choosers

In Net logo, choosers let user to choose a value for a global variable from a list of choices, presented in a drop down menu. The choices may be strings, numbers, Booleans, or lists.

In our simulator, we have used the following switches:

Scenario Chooser - Choosers are basically drop-down menus from where user has to select any one of the choices on display. Scenarios are different models of network implemented in the same world space to test for their effectiveness. We have created four basic scenarios for the user to choose from. However, creating additional scenarios is pretty straightforward once user can think of one. With little programming knowledge, an advanced user is supposed to be capable of doing so. The built-in scenarios in scenario_chooser include basic – involving just one AUV, two AUVs- involving two AUVs one above another in layers, three AUVs –same as two AUVs case but involving an additional AUV in the bottom layer to take care of other half of the nodes and finally two_boats – it is exactly same as three_AUVs scenario but involving 2 boats instead of one, the second boat is considered to be cruising in from opposite direction.

Data Record - The two most important information from simulation are the input i.e. the total volume of data generated by nodes and output i.e. the total data received by the sinks on-shore with respect to time. In the simulator, there are two ways to have this information. If data_record is set to 'on display' mode, these information are shown in monitors and plot viewer in numerical and graphical representations respectively. Whereas by selecting 'file_save' option, the user can have the chance to export information in comma separated values (.csv) format in tabular form. CSV files can be read by Microsoft Excel or Mat lab, where further analysis can be carried out with user's freedom.

Terminate Simulation – As mentioned earlier, there are two ways to terminate a running simulation automatically. Choosing between 'elapsed time' and 'data volume' option, users can stop simulation either by setting simulation time or data volume sliders respectively. The third option is "never" as it signifies the simulation is required to be stopped manually by clicking 'Simulate' button on the interface.

5.2.1.4 Buttons

Net logo defines button as either *once* or *forever*. When user clicks on a once button, it executes its instructions once. The forever button executes the instructions over and over, until user clicks on the button again to stop the action.

In our simulator, we have used the following two buttons:

Setup Button - With the hit of the setup button, blank modeling environment of Net logo gets filled with the desired underwater network scenario. Setup operation goes through many procedures to finally build the world according to user's input in the interface. To name a few, setup button resets the timer, zeros all data count, check for status of real time switch, data record mode, and most importantly models ocean, shore, creates sinks, boat, AUVs, nodes etc. Once setup button is pressed, the simulator is ready to perform simulation on the scenario displayed on-screen.

Simulate Button – This button essentially starts the simulation. Behind this button works many procedures like one that is responsible for moving the AUVs, launching the boat, generating data in each nodes, performing data transfers among linked bodies etc. Simulate button is a *forever* button, pressing this button again would promptly halt the running simulation.

5.2.1.5 Monitors

Net logo describe monitors as display that shows value of any reporter. The reporter could be a variable, a complex reporter, or a call to a reporter procedure. Monitors automatically update several times per second.

In our simulator, we have used the following monitors:

Node generate – A Monitor displays real time value of an assigned variable while the simulation is running. In our simulator, each node is programmed to generate a chunk of bytes every second. The node generate monitor displays the total data generated by each node with the passage of time.

Total node generate – If total number of nodes in the model is greater than one, which is the case in practical networks, the total node generate monitor shows the sum of data generated by the all the nodes in the network combined. This value may be considered as the input of the system. Total node generate = (node generate * number of nodes)

Sink receive – Data transmission is considered successful once it reaches final destination terminal through the intermediaries in the system. In our case, data obtained by either of the sinks is considered as received data, and this can be thought of as output of the system. Therefore, sink receive monitor displays the total data received by the both the sinks combined.

Total node sent – AUV 2 collects data from sensor nodes as it moves along its path. The sum of all data bytes sent by all the nodes combined is displayed in the total node sent monitor.

AUV sent - AUV 1 collects data from AUV 2 when AUV 1-AUV 2 link is existent. The data bytes sent by AUV 2 to AUV 1 above it is displayed in the AUV sent monitor.

AUV1_sent – The boat collects data from AUV 1 when boat-AUV 1 link is existent. The data bytes sent by AUV 1 to boat above it is displayed in the AUV1_sent monitor.

Boat sent – The on-shore sinks accumulates data from boat as the boat moves along its path. The sum of data gathered by both the sinks is displayed on the boat sent monitor.

Total node storage – As previously mentioned, nodes can only transfer data when an AUV is within reach. To keep hold of its unremittingly generated data, nodes requires to store data immediately after generation. The total node storage monitor displays total stored data of all nodes combined.

Timer – The timer is triggered by the hitting the setup button. It shows elapsed time after setup button has been pressed in seconds. This value is vital to keep track of the simulation duration.

5.2.1.6 Plot

Net logo defines plot as a graphical view of data the model is generating.

We have used only one plot window in our simulator.

Total data transfer – The total data transfer window represents two graphs namely total node generate and sink receive on the same time scale. The graphs are plotted in real-time as the simulation progresses.

CHAPTER

6

Data Collection & Result Analysis

6.1 Technical Specification

To make our simulation realistic, we have gone through many technical specifications of practically existing sensor nodes, AUVs etc. We fetched data from the tech sheets and implemented those into simulator.

Given below is a summary of the properties of nodes, AUVs and boat modem,

Nodes:

- Maximum Operating Depth: 200m~1km
- Acoustic Modem Range: 250m~1km
- Bit rate: 25~100bps

Autonomous Underwater Vehicle:

- Maximum Operating Depth: 200m~1km
- Acoustic Modem Range: >1km
- Bit rate: >320kbps
- Speed: 1-4 knots **Boat:**

- Acoustic Modem Range: >1km
- Bit rate: >320kbps
- Speed: 10-25 knots
- Boat to sink connection: Wi-Fi

6.2 Simulation Parameters

For running our simulation, we input the following values of different parameters:

Dimensions:

Length of ocean = 2.3 kilometers

Ocean depth = 1037 meters

Separation between sinks = 1804 meters

Node to node gap = 205 meters

Depth of AUV2 from surface = 837 meters

Depth of AUV1 from surface =537 meters

Speeds:

Speed of AUV1 = 6m/s = 11.66 knots Speed of AUV2 = 5m/s = 9.72 knots

Speed of boat = 15m/s = 29.15 knots

Sensor ranges:

Node sensor range =300 meters

AUV sensor range = 1000 meters

AUV2 sensor range= 500 meters

Boat sensor range = 1000 meters

Others:

Number of nodes = 10 units

Node data generation rate = 5 bytes per second

6.3 Scenario Comparison

Using our simulator we basically created four scenarios to test for their performance and effectiveness.

6.3.1 Scenario 1: One AUV and one boat

In scenario 1, only one AUV and one boat is present for data transfer see Fig. 5(a) and Fig. 5(b). Given that the range of AUV is limited, boat-AUV link exists for only a very short amount of time.

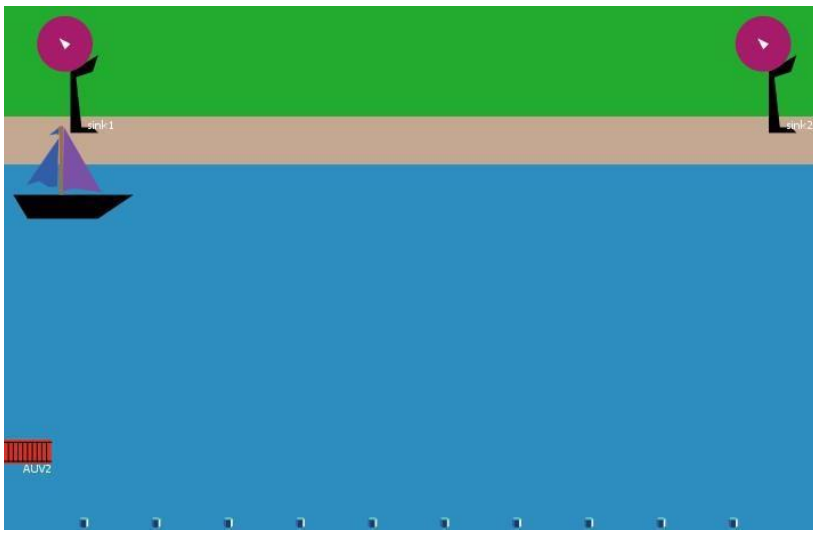

Fig. 5(a) Scenario 1: One AUV and one boat

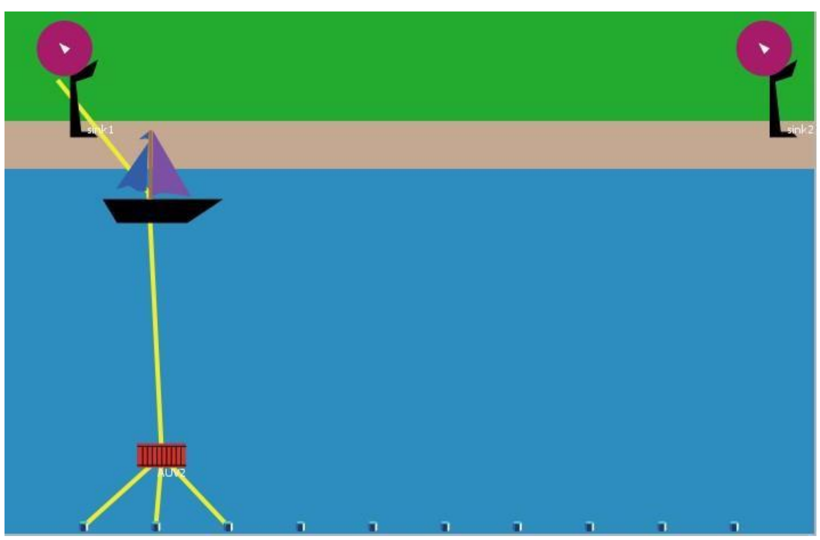

Fig. 5(b) Scenario 1, a moment during simulation, links shown

6.3.2 Scenario 2: Two AUVs and one boat

In scenario 2, two AUVs are in stack format and a boat is present for data transfer. So to compensate for the limited range of AUV acoustic signals we stack two AUVs to move across the entire length of our selected sea bed, see Fig. 5(c) and Fig. 5(d).

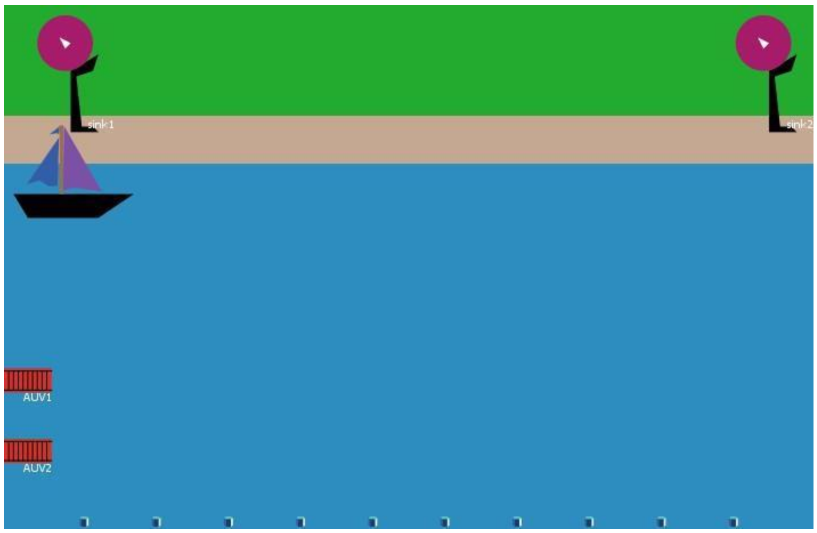

Fig. 5(c) Scenario 2: Two AUVs in layers and one boat

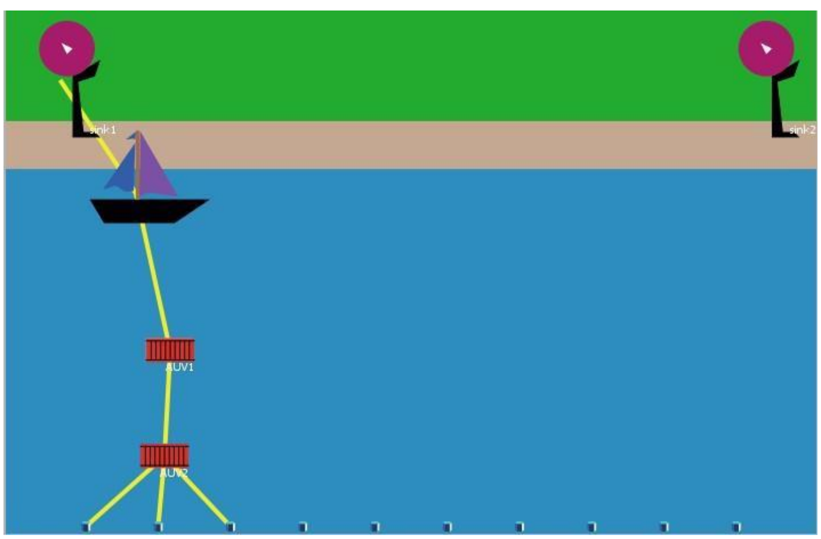

Fig. 5(d) Scenario 2, a moment during simulation, links shown

6.3.3 Scenario 3: Three AUVs and one boat

In the third scenario, two AUVs are present side-by –side in the first layer, with an additional one in stack in the second layer, see Fig. 5(e) and Fig. 5(f). This has been done to increase the amount of communication between the nodes and the AUV, given the fact that speed of AUVs are much slower than boat.

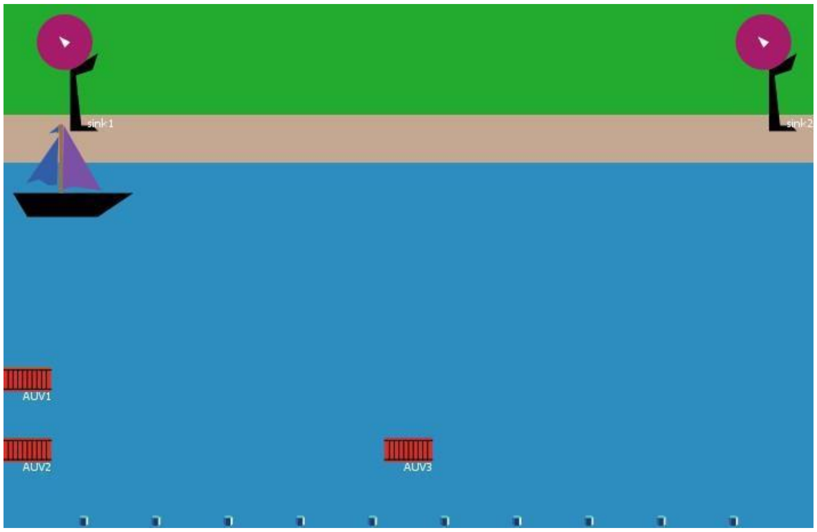

Fig. 5(e) Scenario 3: Three AUVs and one boat

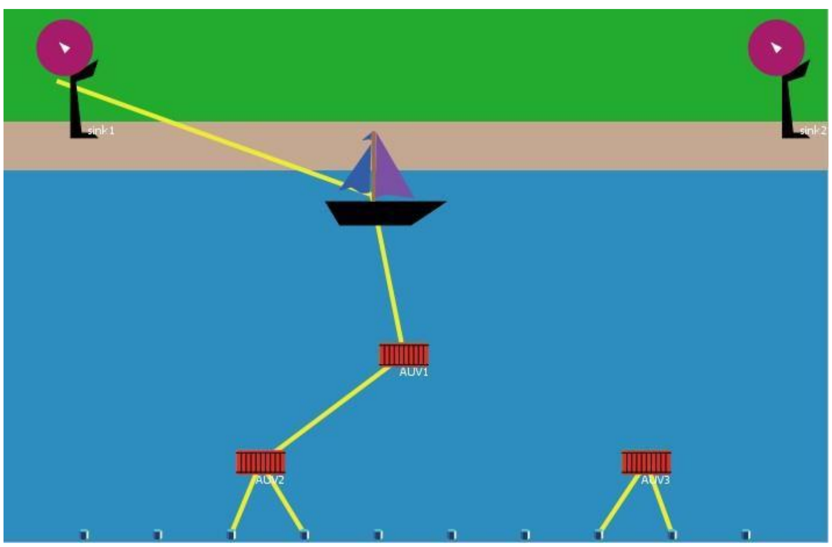

Fig. 5(f) Scenario 3, a moment during simulation, links shown

6.3.4 Scenario 4: Three AUVs and two boats

Scenario 4 is similar in construct to scenario 3, with an additional boat present to increase overall data transfer rate. The second boat is assumed to start its cruise from the opposite end. The idea for this scenario is to increase the amount of communication in the entire system.

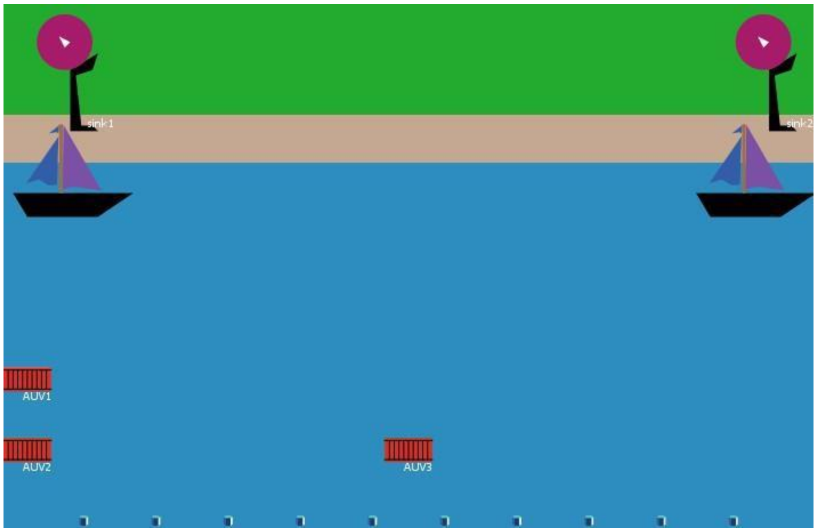

Fig. 5(g) Scenario 4: Three AUVs and two boats

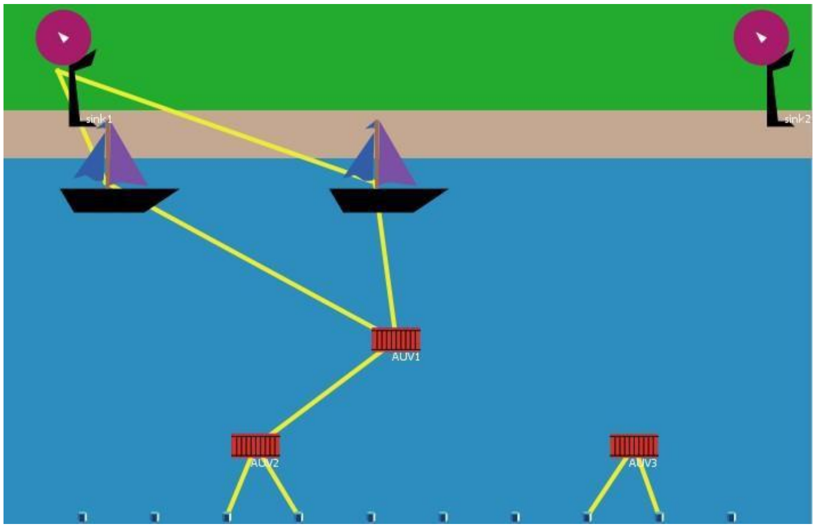

Fig. 5(h) Scenario 4, a moment during simulation, links shown

6.4 Methods of analysis

The 4 different scenarios were compared on the basis of 2 approaches:

- In one approach, the simulation time was kept fixed and the % data received at the sink, the end node, was measured by our simulator for each scenario respectively.
- In another approach, the volume of data received at the sink is kept fixed, and the corresponding time for this data transfer is measured for each scenario.

6.4.1 Constant time analysis

Total data generated by nodes and total data received at sinks are plotted with time. The plots from our simulation for the 4 scenarios for constant time analysis are as follows:

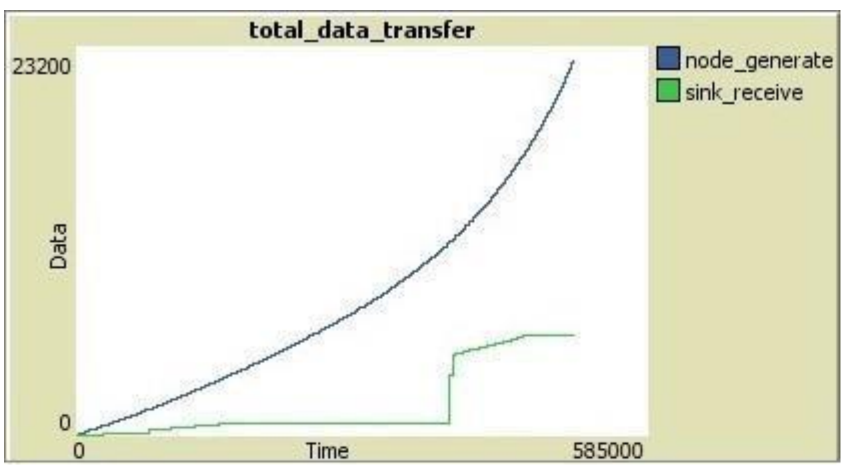

Fig. 5(i) Plot for constant time analysis of scenario 1

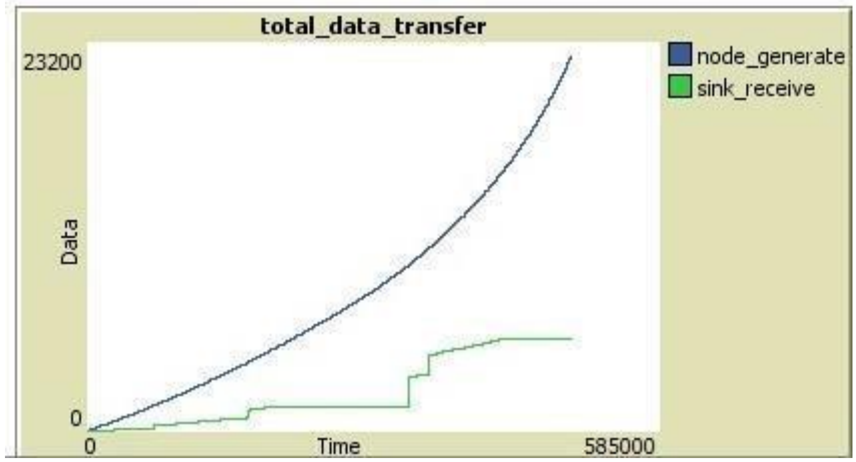

Fig. 5(j) Plot for constant time analysis of scenario 2

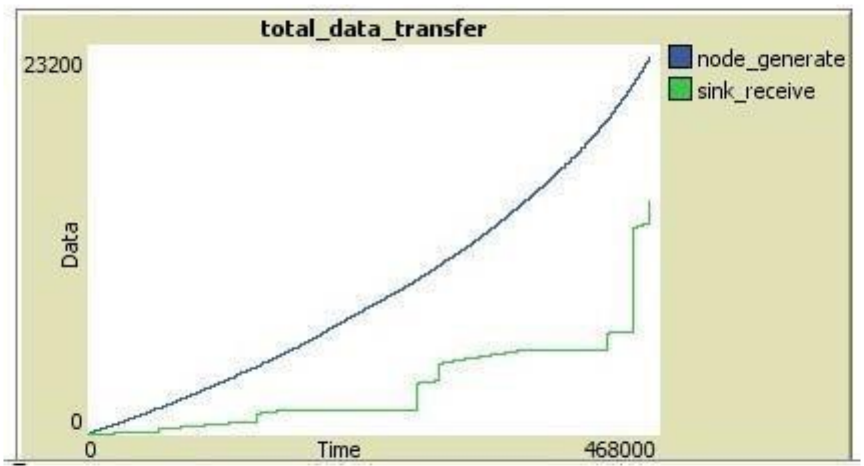

Fig. 5(k) Plot for constant time analysis of scenario 3

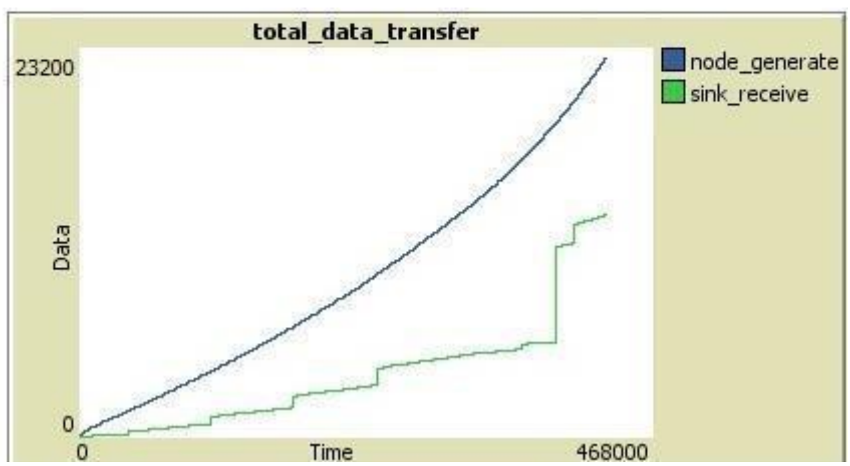

Fig. 5(l) Plot for constant time analysis of scenario 4

In Constant time analysis, for all scenarios, simulation time was considered 8 minutes. Since the simulation time was fixed, amount of total data generated by the nodes was same for all cases. On the basis of different amount of received data, we calculated percentage data transferred for all 4 scenarios as follows:

$$\text{Data Transferred} = \frac{\text{Amount of Data Received at sink node}}{\text{Amount of Data Generated}} \times 100$$

Here, amount of data generated = simulation time (in sec) * number of nodes * data generation rate

= (8 * 60) * 5 * 10 bytes

= 24000 bytes

Scenario 1

Amount of data received at sink node = 5855 bytes

- % Data Transferred = **24.4%**

Scenario 2

Amount of data received at sink node = 6380 bytes

- % Data Transferred = **26.58%**

Scenario 3

Amount of data received at sink node = 14095 bytes

- % Data Transferred = **58.73%**

Scenario 4

Amount of data received at sink node = 14845 bytes

- % Data Transferred = **61.85%**

6.4.2 Constant data analysis

Total data generated by nodes and total data received at sinks are plotted with time. The plots from our simulation for the 4 scenarios for constant data analysis are as follows:

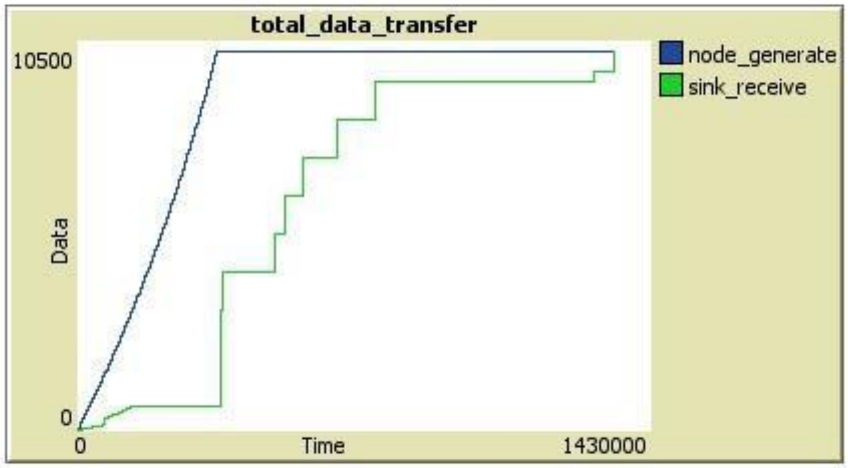

Fig. 5(m) Plot for constant data analysis of scenario 1

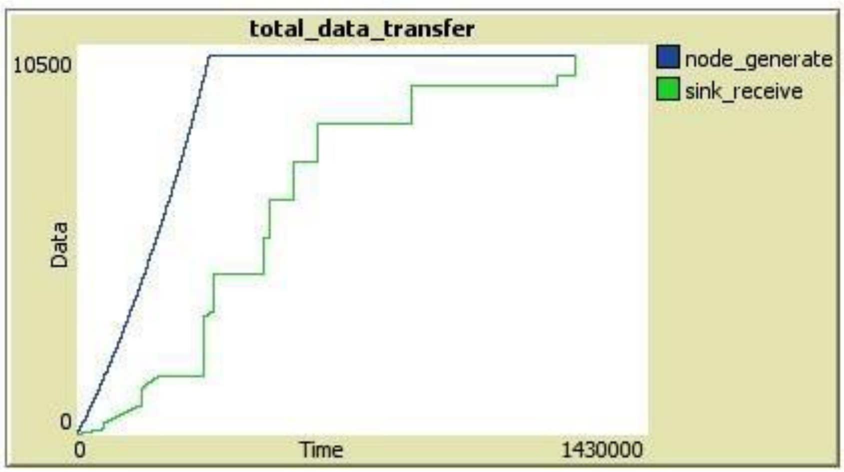

Fig. 5(n) Plot for constant data analysis of scenario 2

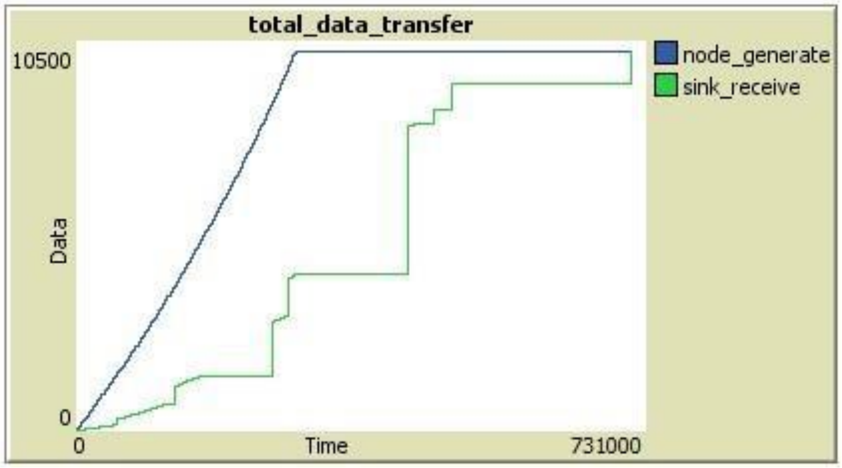

Fig. 5(o) Plot for constant data analysis of scenario 3

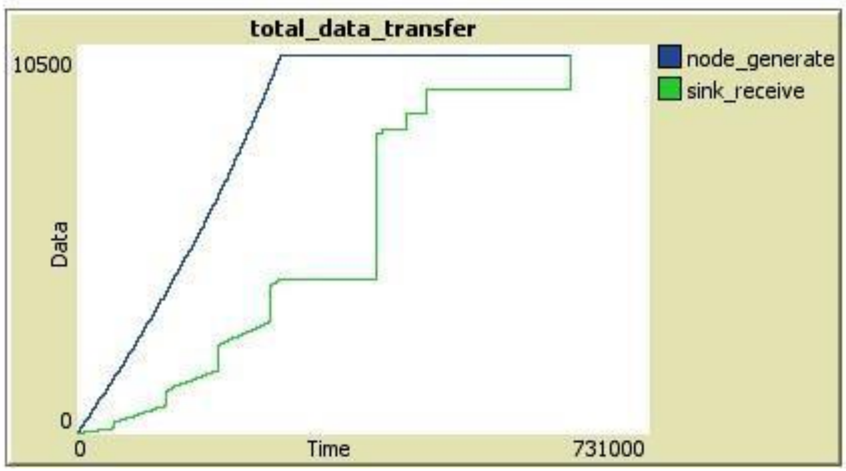

Fig. 5(p) Plot for constant data analysis of scenario 4

In Constant data analysis, for all scenarios, volume of data for generation and transfer was considered 10 kilobytes. Since the data generation was fixed, nodes stopped generating when combined node generation reached precisely

10240 bytes i.e. 10 kilobytes. We therefore measured the time required for data transfer for each scenarios.

$$\text{Time taken for data generation} = \frac{\text{Amount of data generated}}{\text{Number of nodes} * \text{Data generation rate}}$$
$$= \frac{10240}{10 * 5}$$
$$= 204.8 \text{ sec}$$

Scenario 1

Total transfer time = 2166.743 sec = 36.112 minutes

- Time taken after generation completion = (2166.743 - 204.8) = **1961.943 sec**

Scenario 2

Total transfer time = 1891.045 sec = 31.517 minutes

- Time taken after generation completion = (1891.045 - 204.8) = **1686.245 sec**

Scenario 3

Total transfer time = 1234.04 sec = 20.567 minutes

- Time taken after generation completion = (1234.040 - 204.8) = **1029.240 sec**

Scenario 4

Total transfer time = 1240.561 sec = 20.676 minutes

- Time taken after generation completion = (1240.561 - 204.8) = **1035.761 sec**

5.5 Locating the optimum point

To locate the optimum point we use both the analysis of constant time and constant data. Using the constant time analysis we find that the amount of data transferred within a set period of 8 minutes between the sink node and the sensor nodes is highest in scenario 3 and in scenario 4.

Then we keep the total amount of data constant at 10 kilobytes and then run the simulation in the constant data analysis to find that the least amount of time required to transfer data between the nodes and sink is in scenario 3 as well followed by scenario 4.

After this we compare the scenarios from the time constant analysis by comparing the increasing of data transfer whenever we move from one scenario to the next by addition of different components to the network.

The following chart Fig. 5(q) shows a comparison of the four scenarios in terms of increase in percentage data transferred.

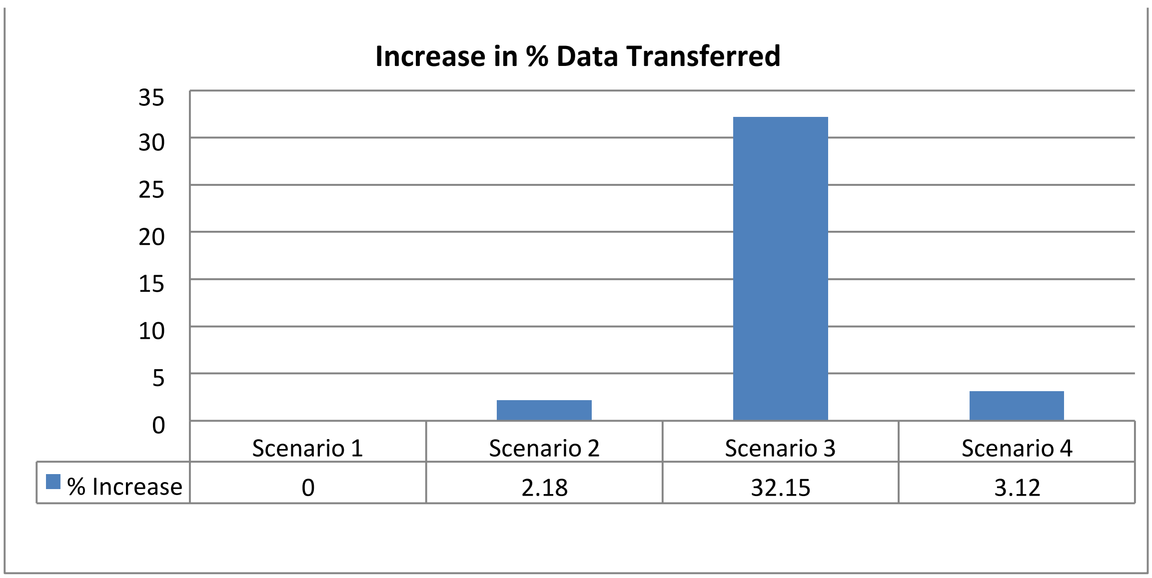

Fig. 5(q) Comparison of four scenarios

Highest increase in percentage data transferred is observed at scenario 3.

Thus we select **scenario 3** of three AUVs and one boat as the optimum point.

CHAPTER

7

Conclusion

Conclusion

DTN is a technology that is very new to the world of networking. Underwater data is an essential part of everyday life of human beings and every year billions of dollars are being spent to recover as much information as possible.

In the first part we analyzed different Routing protocol and did the comparative studies. We came to conclusion that DBR has the both high performance and high reliability. So in most of the cases DBR gives us the advantage and most suitable for Underwater environment but in few other cases depending on the demand other routing protocol can give decent performance As each of the protocol has its own feature.

The thesis exploited this vast field of data availability and aimed to achieve the efficient point at which the data transfer would be done at maximum for the money being spent behind the projects.

The project can be implemented in different scenarios and can be modified according to the different requirements of the underwater terrain at different locations.

Furthermore, we tried to reach the optimum point of operation simulating two different sorts of operation to the four scenarios that we selected to understand the project even better.

References

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257 – 279, 2005.

[2] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, "Research challenges and applications for underwater sensor networking," in *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.*, vol. 1. IEEE, 2006, pp. 228–235.

[3] 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, no. 6, pp. 1908–1927, 2011.

[4] J. G. Proakis, E. M. Sozer, J. A. Rice, and M. Stojanovic, "Shallow water acoustic networks," *IEEE communications magazine*, vol. 39, no. 11, pp. 114–119, 2001.

[5] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE journal of oceanic engineering*, vol. 25, no. 1, pp. 72 – 83, 2000.

[6] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *Ieee Network*, vol. 20, no. 3, pp. 12–18, 2006.

[7] J. Kong, J.-h. Cui, D. Wu, and M. Gerla, "Building underwater adhoc networks and sensor networks for large scale real-time aquatic applications," in *MILCOM 2005-2005 IEEE Military Communications Conference*. IEEE, 2005, pp. 1535–1541.

[8] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 23–33, 2007.

[9] I. F. Akyildiz and E. P. Stuntebeck, "Wireless underground sensor networks: Research challenges," *Ad Hoc Networks*, vol. 4, no. 6, pp. 669–686, 2006.

[10] K. Ovaliadis, N. Savage, and V. Kanakaris, "Energy efficiency in underwater sensor networks: a research review," *Journal of Engineering Science and Technology Review (JESTR)*, vol. 3, no. 1, pp. 151–156 , 2010.

[11] M. C. Domingo and R. Prior, "Energy analysis of routing protocols for underwater wireless sensor networks," *Computer communications*, vol. 31, no. 6, pp. 1227–1238, 2008.

[12] D. Pompili and I. F. Akyildiz, "Overview of networking protocols for underwater wireless communications," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 97–102, 2009.

[13] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 34 – 43, 2007.

[14] L. Lanbo, Z. Shengli, and C. Jun-Hong, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 8, pp. 977–994 , 2008.

[15]M. Erol and S. Oktug, “A localization and routing framework for mobile underwater sensor networks,” in *INFOCOM Workshops 2008, IEEE*. IEEE, 2008, pp. 1–3.
[16]Z. Zhou, J.-H. Cui, and S. Zhou, “Efficient localization for large-scale underwater sensor networks,” *Ad Hoc Networks*, vol. 8, no. 3, pp. 267– 279, 2010.
[17]P. Xie, J.-H. Cui, and L. Lao, “Vbf: vector-based forwarding protocol for underwater sensor networks,” in *International Conference on Research in Networking*. Springer, 2006, pp. 1216–1221.
[18]I. F. Akyildiz, D. Pompili, and T. Melodia, “Challenges for efficient communication in underwater acoustic sensor networks,” *ACM Sigbed Review*, vol. 1, no. 2, pp. 3–8, 2004.
[19]J.-E. Garcia, “Ad hoc positioning for sensors in underwater acoustic networks,” in *OCEANS’04. MTTS/IEEE TECHNO-OCEAN’04*, vol. 4. IEEE, 2004, pp. 2338–2340.
[20]V. Chandrasekhar and W. Seah, “An area localization scheme for underwater sensor networks,” in *OCEANS 2006-Asia Pacific*. IEEE, 2007, pp. 1 – 8.
[21]V. Chandrasekhar, W. K. Seah, Y. S. Choo, and H. V. Ee, “Localization in underwater sensor networks: survey and challenges,” in *Proceedings of the 1st ACM international workshop on Underwater networks*. ACM, 2006, pp. 33–40.
[22]M. Erol, L. F. Vieira, and M. Gerla, “Localization with dive’n’rise (dnr) beacons for underwater acoustic sensor networks,” in *Proceedings of the second workshop on Underwater networks*. ACM, 2007, pp. 97–100.
[23]K. Chen, Y. Zhou, and J. He, “A localization scheme for underwater wireless sensor networks,” *International Journal of Advanced Science and Technology*, vol. 4, 2009.
[24]M. Ayaz, L. T. Jung, A. Abdullah, and I. Ahmad, “Reliable data deliveries using packet optimization in multi-hop underwater sensor networks,” *Journal of King Saud University-Computer and Information Sciences*, vol. 24, no. 1, pp. 41–48, 2012.
[25]M. A. Rahman, A. El Saddik, and W. Gueaieb, “Wireless sensor network transport layer: State of the art,” in *Sensors*. Springer, 2008, pp. 221 – 245.
[26]B. Scheuermann, C. Lochert, and M. Mauve, “Implicit hop-by-hop congestion control in wireless multihop networks,” *Ad Hoc Networks*, vol. 6, no. 2, pp. 260–286, 2008.
[27]R. de Oliveira and T. Braun, *TCP in wireless mobile ad hoc networks*. Universitat¨ Bern. Institut fur¨ Informatik und Angewandte Mathematik, 2002.

[28]W. S. Broecker, T.-H. Peng, and Z. Beng, *Tracers in the Sea*. LamontDoherty Geological Observatory, Columbia University, 1982.

[29]D. Shin, D. Hwang, and D. Kim, "Dfr: an efficient directional floodingbased routing protocol in underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 17, pp. 1517–1527 , 2012.

[30]S. N. Hussain, M. M. Hussain, and M. G. Aslam, "Flooding based routing techniques for efficient communication in underwater wsns1," *European Journal of Computer Science and Information Technology*, vol. 3, no. 3, pp. 52–60, 2015.

[31]M. C. Domingo and R. Prior, "A distributed clustering scheme for underwater wireless sensor networks," in *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*.
IEEE, 2007, pp. 1 – 5.

[32]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*. ACM, 2008 , pp. 75–82.

[33]J. M. Jornet and M. Stojanovic, "Distributed power control for underwater acoustic networks," in *OCEANS 2008*. IEEE, 2008, pp. 1 – 7.

[34]H. Yan, Z. J. Shi, and J.-H. Cui, "Dbr: depth-based routing for underwater sensor networks," in *International Conference on Research in Networking*. Springer, 2008, pp. 72–86.

[35] En.wikipedia.org. 2013. *Delay-tolerant networking*. [online] Available at: http://en.wikipedia.org/wiki/Delay-tolerant_networking [Accessed: 8 Oct 2013].

[36] En.wikipedia.org. 2013. *Underwater acoustic communication*. [online] Available at:
http://en.wikipedia.org/wiki/Underwater_acoustic_communication
[Accessed: 8 Oct 2013].

[37] K. Fall. A delay-tolerant network architecture for challenged internets, in: Proceedings of ACM SIGCOMM'03, August 2003.

[38] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," Computer Networks (Elsevier) Journal, vol. 38, no. 4, pp. 393–422, Mar. 2002.

[39] The Delay Tolerant Networking Research Group, Available from <http://www.dtnrg.org>.

[40] Heidemann, J., Li, Y., Syed, A., Wills, J., and Ye, W. Underwater sensor networking: Research challenges and potential applications. Tech. Rep. ISI-TR-2005-603, USC/Information Sciences Institute, July 2005.

[41] M. J. Khabbaz, C. M. Assi, and W. F. Fawaz, "Disruption-tolerant networking: A comprehensive survey on recent developments and persisting challenges," IEEE Communications Surveys & Tutorials, vol. PP, no. 99, 2011, pp. 1–34.

[42] Ccl.northwestern.edu. 2013. *NetLogo Home Page*. [online] Available at: http://ccl.northwestern.edu/netlogo/ [Accessed: 8 Oct 2013].