## ABSTRACT

For energy-efficient resource management, void node avoidance is one of the key objectives in the energy constrained underwater wireless sensor networks (UWSNs). In this work, we propose two new routing protocols for the UWSN which is one of the end parts of a cloud. The first protocol is avoiding void node with adaptive hop-by-hop vector based forwarding (AVN-AHH-VBF), and the second is cooperation- based AVN-AHH-VBF (CoAVN-AHH-VBF). In both schemes, sensor nodes forward data packets in multi- hop fashion within a virtual pipeline. The nodes outside the pipeline do not forward data packets to avoid flooding in the network. At each hop, forwarding toward void region of the network is avoided by utilizing two hop information. Results of extensive simulations show that our proposed schemes significantly improve the network performance in terms of delivery ratio, energy expenditure and delay as compared with the selected existing scheme (AHH-VBF).

# CHAPTER-1

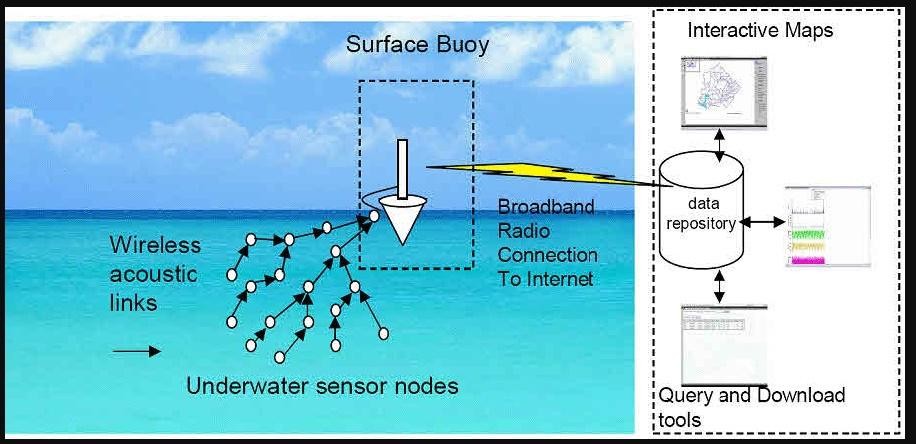
INTRODUCTION

## WIRELESS NETWORK

A Wireless sensor network spatially distribute autonomous sensors to monitor physical or environment conditions,(such as sound ,pressure, etc.) and to cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling control of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

## Under Water Sensor Networks

UWSN consists of sensor nodes equipped with acoustic modems and a sink node equipped with both acoustic and radio modems. These networks are used for monitoring rivers, lakes and oceans. Application examples include oceanographic data collection, oil spill monitoring, military training, tactical surveillance, disasters prevention, undersea pollution monitoring, submarine detection, aquatic habitat monitoring, etc. Generally, acoustic waves are used for underwater communication. However, the detrimental nature of acoustic channel leads to high bit error rate (BER), low bandwidth, high propagation delay, etc. These challenges lead to high energy consumption of the network nodes, and low reliability of the received data.



Random deployment of nodes allows for some part of the underwater network area to be less populated (low node density), while leaving other parts more populated (high node density). This deployment technique leaves network areas prone to void holes. In UWSNs, generally data packets travel from the bottom anchored source nodes to the on surface sink nodes. Thus, the sensor nodes near the sink are heavily engaged in data transmission which results in the void regions near the sink. In such situations, no more data packets can be successfully delivered to the sink resulting in large amount of energy loss. Wireless system designers enhance reliability of the transmitted data by taking into account diverse techniques; such as, the number of duplicated data packets at the receiver node. To achieve spatial diversity, Multiple- Input Multiple-Output (MIMO) and multi node cooperative Transmission are efficient approaches. However, the former approach requires hardware at each node with higher complexity and cost. The later approach is accomplished via cooperative routing where multiple nodes are exploited to transmit/relay duplicated data packets arriving at the destination after some delay.

## Limitations of Wireless Sensor Networks

1. Possess very little storage capacity – a few hundred kilobytes
2. Possess modest processing power-8MHz
3. Works in short communication range – consumes a lot of power
4. Requires minimal energy – constrains protocols
5. Have batteries with a finite life time
6. Passive devices provide little energy

### CHALLENGES IN UNDERWATER WIRELESS SENSOR NETWORKS

The challenging issues of the Underwater Networks, and then proceed to communications and networking layers, followed by a discussion on various security issues of network layer. Topology: High dynamic due to the continual movement of sensor nodes by the current movement of the water. Communication Media: Acoustic waves for underwater environment and radio waves for water surface. Position information: Unavailable by GPS, because GPS uses high frequency waves which are rapidly absorbed in the sea water. Network Components: Underwater ordinary nodes, sinks, AUV, and onshore base station. Range: Usually used in vast ocean areas. Speed of Medium: Acoustic Velocity in water is about 1500 m/s

# CHAPTER-2

## LITERATURE SURVEY

* 1. **VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks**

UWSNs are significantly different from terrestrial sensor networks in the following aspects: low bandwidth, high latency, node float mobility (resulting in high network dynamics), high error probability, and 3-dimensional space. These new features bring many challenges to the network protocol design of UWSNs. In this paper, we propose a novel routing protocol, called vector-based forwarding (VBF), aiming to provide robust, scalable and energy efficient routing. VBF is essentially 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. Moreover, packets are forwarded in redundant and interleaved paths, which add robustness to VBF. Further, we develop a localized and distributed self-adaptation algorithm, which helps to enhance the performance of VBF. The self-adaptation algorithm allows the nodes to weigh the benefit to forward packets and reduce energy consumption by discarding the low benefit packets. We evaluate the performance of VBF through extensive simulations. Our experiment results show that for networks with small or medium node mobility (1 m/s-3 m/s), VBF can effectively accomplish the goals of robustness, energy efficiency, and high success of data delivery.

## Improving the Robustness of Location-Based Routing for

## Underwater Sensor Networks

This paper investigates a fundamental networking problem in underwater sensor networks: robust and energy efficient routing. We present an adaptive location-based routing protocol, called hop-by-hop vector-based forwarding (HH- VBF). It uses the notion of a “routing vector” (a vector from the source to the sink) acting as the axis of the “routing pipe”, similar to the vector based forward (VBF) routing . Unlike the original VBF approach, however, HH-VBF suggests the use of a routing vector for each individual forwarder in the network, instead of a single network-wide source-to-sink routing vector. By the creation of the hop-by-hop vectors, HH-VBF can overcome two major problems in VBF: (1) too small data delivery ratio for sparse networks; (2) too sensitive to “routing pipe” radius threshold. We conduct simulations to evaluate HH-VBF, and the results show that HH-VBF yields much better performance than VBF in sparse networks. In addition, HH-VBF is less sensitive to the routing pipe radius threshold. Furthermore, we also analyze the behavior of HH-VBF and show that assuming proper redundancy and feedback techniques, HH-VBF can facilitate the avoidance of any “void” areas in the network.

## A Distributed Data Gathering Protocol Using

## Underwater Sensor Networks

A distributed data-gathering scheme using an autonomous underwater vehicle (AUV) working as a mobile sink to gather data from a randomly distributed underwater sensor network where sensor nodes are clustered around several cluster headers. Unlike conventional data-gathering schemes where the AUV visits either every node or every cluster header, the proposed scheme

allows the AUV to visit some selected nodes named path-nodes in a way that reduces the overall transmission power of the sensor nodes. Monte Carlo simulations are performed to investigate the performance of the proposed scheme compared with several preexisting techniques employing the AUV in terms of total amount of energy consumption, standard deviation of each node’s energy consumption, latency to gather data at a sink, and controlling overhead. Simulation results show that the proposed scheme not only reduces the total energy consumption but also distributes the energy consumption more uniformly

over the network, there by increasing the lifetime of the network.

## Delay-Sensitive Routing Schemes for Underwater

## Acousti Sensor Networks

Underwater Acoustic Sensor Networks (UASNs) offer their practicable applications in seismic monitoring, sea mine detection, and disaster prevention. In these networks, fundamental difference between operational methodologies of routing schemes arises due to the requirement of time-critical applications; therefore, there is a need for the design of delay-sensitive techniques. In this paper, Delay-Sensitive Depth-Based Routing (DSDBR), Delay-Sensitive Energy Efficient Depth-Based Routing (DSEEDBR), and Delay Sensitive Adaptive Mobility of Courier nodes in Threshold-optimized Depth-based routing (DSAMCTD) protocols are proposed to empower the depth-based routing schemes. The performance of the proposed schemes is validated in UASNs. All of the three schemes formulate delay-efficient Priority Factors (PF) and Delay-Sensitive

Holding time (DSH T) to minimize end-to-end delay with a small decrease in

network throughput. These schemes also employ an optimal weight function (W唨)

for the computation of transmission loss and speed of received signal.

Furthermore, solution for delay lies in efficient data forwarding, minimal relative transmissions in low-depth region, and better forwarder selection. Simulations are performed to assess the proposed protocols and the results indicate that the three schemes largely minimize end-to-end delay along with improving the transmission loss of network.

### ENERGY EFFICIENT DATA TRANSMISSION IN UNDERWATER ACOUSTIC SENSOR NETWORK (UW-ASN) USING CLUSTER HEAD SELECTION PROTOCOL

UWSN will find applications in data collection, oceanographic, pollution monitoring, offshore exploration, disaster prevention and tactical surveillance. Underwater wireless sensor networks consist of a number of sensors and nodes that interact to collect data and perform collaborative tasks. Designing energy-efficient routing protocols for this type of networks is essential and challenging because sensor nodes are powered by batteries, and are difficult to replace or recharge, because underwater communications are severely affected by network dynamics .The goal of this paper is to increase energy consumption of the network with cluster head selection algorithm. Cluster head selection process will be performed with particle swarm optimization approach.

# CHAPTER-3

### SYSTEM ANALYSIS 3.1 ARCHITECTURE DIAGRAM

Here source file is uploaded and received by router. It problems finding the shortest path and encryption of file receives decrypt the file for viewing of contents.

* 1. **EXISTING SYSTEM**

The most of the related works preclude the ambiguity of the environment by assuming that malicious dropping is the only source of packet loss, so that there is no need to account for the impact of link errors. On the other hand, for the small number of works that differentiate between link errors and malicious packet drops, their detection algorithms usually require the number of maliciously-dropped packets to be significantly higher than link errors, in order to achieve an acceptable detection accuracy.

Depending on how much weight a detection algorithm gives to link errors relative to malicious packet drops, the related work can be classified into the following two categories.

* + - The first category aims at high malicious dropping rates, where most (or all) lost packets are caused by malicious dropping.
    - The second category targets the scenario where the number of maliciously dropped packets is significantly higher than that caused by link errors, but the impact of link errors is non-negligible.

DISADVANTAGES OF EXISTING SYSTEM

 More Packet drops.

 Difficult to find void nodes in under water.  No more security in the existing system.

### PROPOSED SYSTEM

* In the proposed system, the system proposes two new routing schemes for UWSNs: (i) co-operative CoAVN-AHH-VBF and (ii) non- co-operative AVN-AHH-VBF. The later scheme has also been analyzed with the Bit Error Rate (BER); so is given the name AVN-AHH-VBF-B. Both the schemes avoid the void node by checking the status of a node before transmitting the data packet (using two hop information).
* The proposed schemes efficiently select forwarder nodes on the bases of least depth in the pipeline and regions towards destination (RTD) in the range of the source node.

The proposed system also modifies the holding time equation by finding the number of hops to be traversed and the number of neighbors of a source node in the network. Simulation results show that the proposed schemes perform better than the selected existing scheme in terms of the selected performance metrics.

### ADVANTAGES OF PROPOSED SYSTEM:

* The proposed is more efficient due to Secure protocols for energy efficiency.
* The system is more secure due to lack of packet drops avoiding techniques.

# CHAPTER-4

### SYSTEM SPECIFICATION

**Hardware Requirements:**

* System : Pentium IV 3.5 GHz.
* Hard Disk : 40 GB.
* Monitor : 14’ Colour Monitor.
* Mouse : Optical Mouse.
* Ram : 1 GB

.

### Software Requirements:

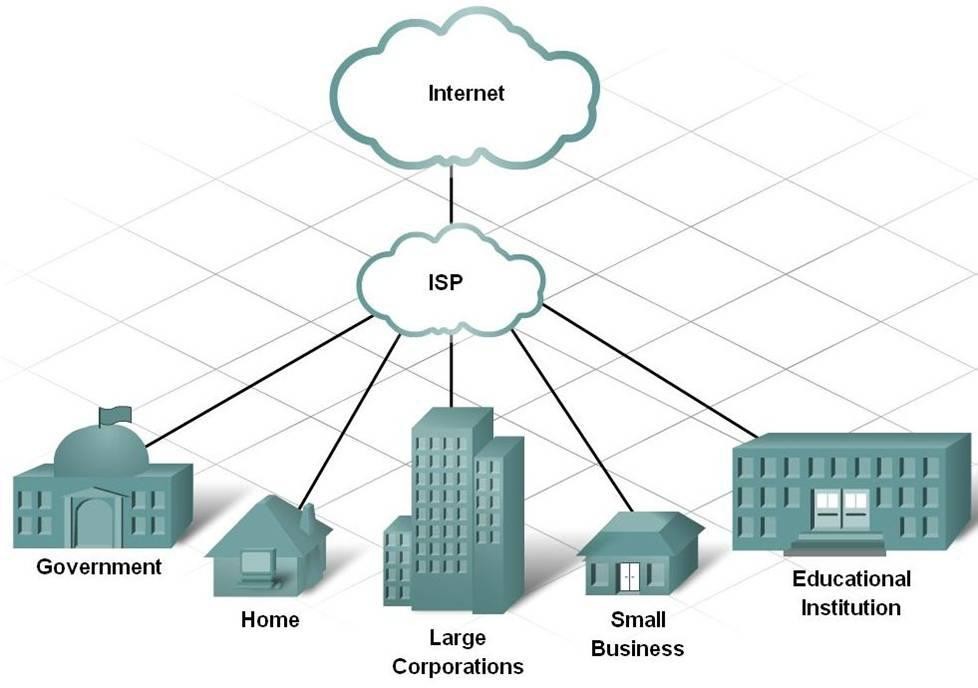
* Operating system : Windows XP or Windows 7, Windows 8.
* Coding Language : Java – AWT,Swings,Networking
* Data Base : My Sql / MS Access.
* Documentation : MS Office
* IDE : Eclipse Galileo
* Development Kit : JDK 1.6

# CHAPTER-5

### MODULES DESCRIPTION

**Service Provider:**

In this module, the service provider browses the file and sends to the particular end users via router. And also service provider can assign energy and assign distances for the nodes in router.ISP literally means Internet service provider or provider. It is a service (most of the time paid for) which allows you to connect to the Internet.



.

Unless you have a specialized line (other than a telephone line), you cannot connect directly to the internet using your telephone line. Indeed, the telephone line was not designed for this: it was originally designed to transport "voice", i.e. a frequency modulation in the range of the voice tone telephone servers only know how to start a conversation from a telephone number unless you resort to a special service, generally it is not possible to have communication between more than two

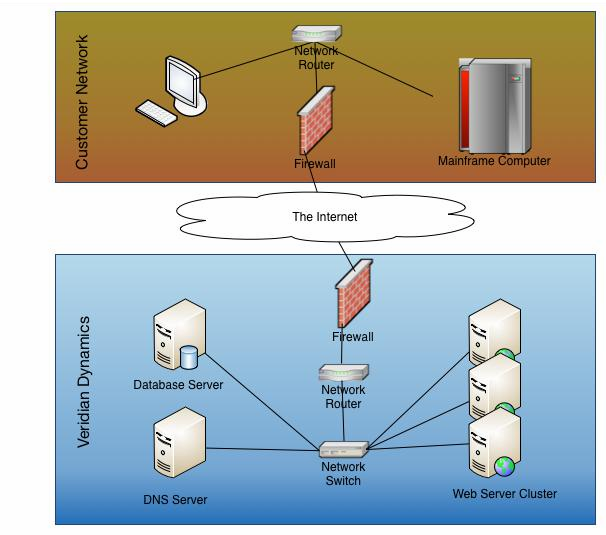
points... So, the internet service provider is an intermediary (connected to the internet by specialized lines) which gives you access to the Internet, using a number which you enter using your modem, and which enables a connection to be established.

Communication between you and the service provider is established according to the PPP protocol which is characterised by: a telephone call initialization of communication verification of the user name (login or userid) verification of the password.

### Router:

In this module, the router sends the file from source to destination (from service provider to end users) by selecting shortest distances between two nodes & sufficient node energy. And if node has less energy than file size then packet dropper in router drops the some packets from file and sends remaining file to the destination. And it can also do some operations like view distances, view energy, view files, view attackers, verify, refresh.

A router is connected to two or more data lines from different networks. When a data packet comes in on one of the lines, the router reads the [network address](https://en.wikipedia.org/wiki/Network_address) information in the packet to determine the ultimate destination. Then, using information in its [routing table](https://en.wikipedia.org/wiki/Routing_table) or [routing policy](https://en.wikipedia.org/wiki/Routing_policy), it directs the packet to the next network on its journey.



The most familiar type of routers are [home and small office routers](https://en.wikipedia.org/wiki/Home_router) that simply forward [IP packets](https://en.wikipedia.org/wiki/IP_packet) between the home computers and the Internet. An example of a router would be the owner's cable or DSL router, which connects to the Internet through an [Internet service provider](https://en.wikipedia.org/wiki/Internet_service_provider) (ISP). More sophisticated routers, such as enterprise routers, connect large business or ISP networks up to the powerful [core routers](https://en.wikipedia.org/wiki/Core_router) that forward data at high speed along the [optical fiber](https://en.wikipedia.org/wiki/Optical_fiber) lines of the [Internet backbone](https://en.wikipedia.org/wiki/Internet_backbone). Though routers are typically dedicated hardware devices, software-based routers also exist.

### Routing different networks

Routers are also often distinguished on the basis of the network in which they operate. A router in a [local area network](https://en.wikipedia.org/wiki/Local_area_network) (LAN) of a single organisation is called an interior router. An exterior router directs packets between hosts in one LAN and hosts in another LAN. A router that is operated in the [Internet](https://en.wikipedia.org/wiki/Internet) backbone is described as exterior router. While routers that connect a LAN with the [wide](https://en.wikipedia.org/wiki/Wide_area_network) area network (WAN) are called border routers, or gateway routers.

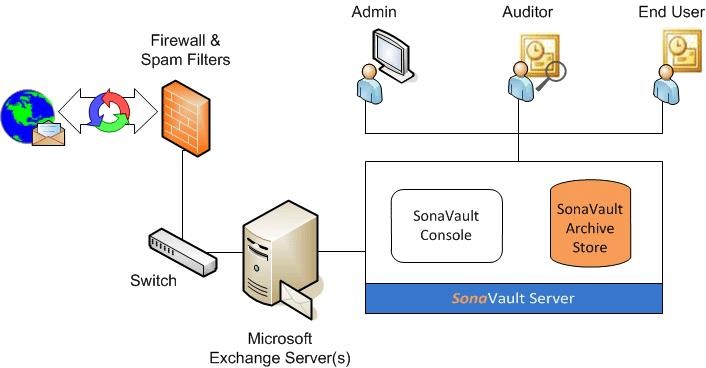
### Auditor:

The auditor discovers the traffic pattern, means it stores the details of dropped packets. It contains details of in which node packets are dropped, how many no of packets dropped, from which file dropped & status of packets.

Network auditing is the collective measures done to analyze, study and gather data about a network with the purpose of ascertaining its health in accordance with the network/organization requirements.

Network auditing primarily provides insight into how effective network control and practices are, i.e. its compliance to internal and external network policies and regulations.

The data is gathered, vulnerabilities and threats are identified, and a formal audit report is sent to network administrators.



Network auditing works through a systematic process where a computer network is analyzed for:

* + - Security
    - Implementation of control
    - Availability
    - Management
    - Performance

It is generally done by an information system auditor, network analyst/auditor or any other individual with a network management and/or security background. It uses both manual and automated techniques to gather data and review network posture. It reviews:

Each node of a network

* + - Network control and security processes
    - Network monitoring processes

Although a network audit may focus more on network control and security, it also reviews processes and measures that ensure network availability, performance and quality of service.

### Destination (End User ):

In this module, there are n no of destinations (A, B, C….). These end users only receive the file from service provider via router. While getting the file from service provider there may be chances of packets dropping, if packets are dropped then end user will gets dropped packets from point to point manager. The end users receive the file by without changing the File Contents. Users may receive particular data files within the network only.

### Attacker:

Attacker is one who makes changes the energy of particular nodes in router. And all attackers’ details stored in router with their all details such as attacker Ip address, attacked node, modified energy and attacked time.

Types of attack:

### ACTIVE ATTACK

 Spoofing:

When a malicious node miss-present his identity, so that the sender change the topology.

 Modification :

When malicious node performs some modification in the routing route, so that sender sends the message through the long route. This attack cause communication delay occurred between sender and receiver.

 Denial of services:

In denial of services attack, malicious node sending the message to the node and consume the bandwidth of the network. The main aim of the malicious node is to be busy the network node. If a message from unauthenticated node will come, then receiver will not receive that message because he is busy and beginner has to wait for the receiver response.

 Sybil:

This attack related to the multiple copies of malicious nodes. The Sybil attack can be happen due to malicious node shares its secret key with other malicious nodes. In this way the number of malicious node is increased in the network and the probability of the attack is also increases.

### PASSIVE ATTACK

 Traffic analysis:

In the traffic analysis attack, an attacker tries to sense the communication path between the sender and receiver. An attacker can found the amount of data which is travel from the route of sender and receiver. There is no modification in data by the traffic analysis.

 Eavesdropping:

This is a passive attack, which occurred in the mobile ad-hoc network. The main aim of this attack is to find out some secret or confidential information from communication. This secrete information may be privet or public key of sender or receiver or any secrete data.

# CHAPTER-6

**Software Environment**

**Java Technology**

Java technology is both a programming language and a platform.

# The Java Programming Language

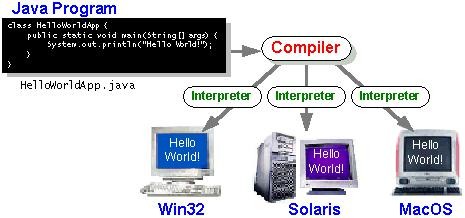
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

* + Simple
  + Architecture neutral
  + Object oriented
  + Portable
  + Distributed
  + High performance
  + Interpreted
  + Multithreaded
  + Robust
  + Dynamic
  + Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called *Java byte codes* the platform-independent codes interpreted by the interpreter on the Java platform.

The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it’s a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



## The Java Platform

A platformis the hardware or software environment in which a program runs. We’ve already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it’s a software-only platform that runs on top of other hardware- based platforms.

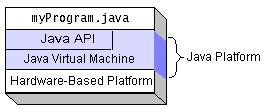
The Java platform has two components:

* The *Java Virtual Machine* (Java VM)
* The *Java Application Programming Interface* (Java API)

You’ve already been introduced to the Java VM. It’s the base for the Java platform and is ported onto various hardware-based platforms.

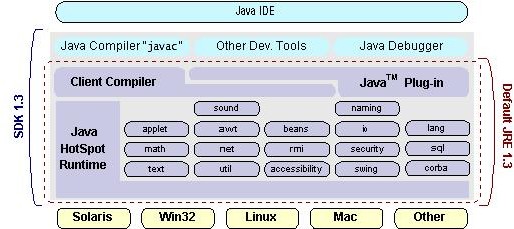
The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that’s running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



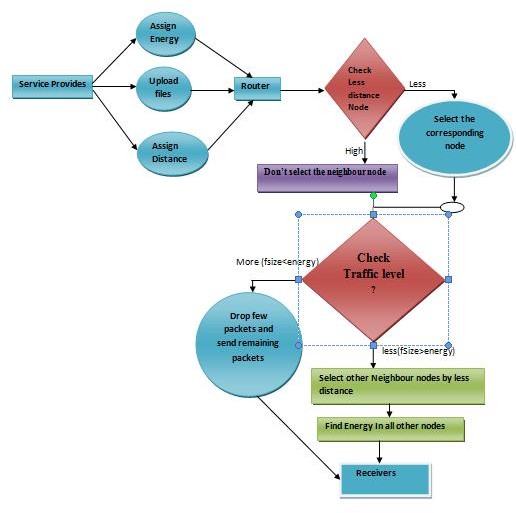
Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



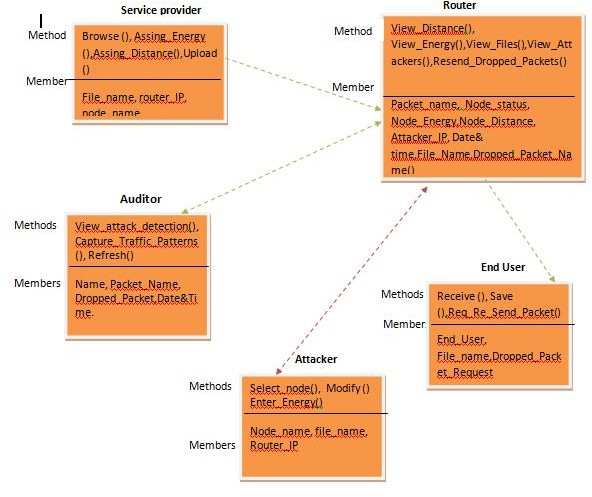
# CHAPTER-7 UML DIAGRAMS

## FLOW CHART



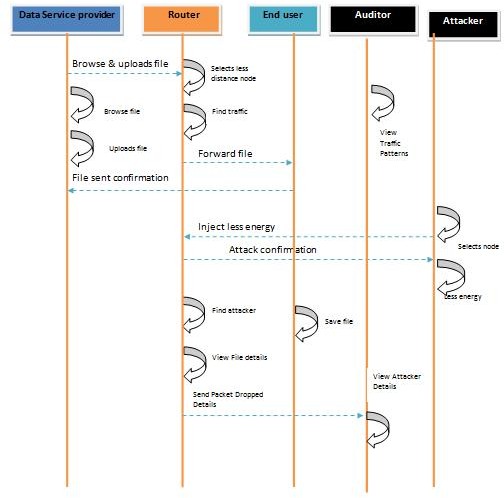
Flow chart defines traveling of file from source to receives. It also checks the traffic level of node. If traffic exists ,it drops the packet and then routed to receives.

### CLASS DIAGRAM



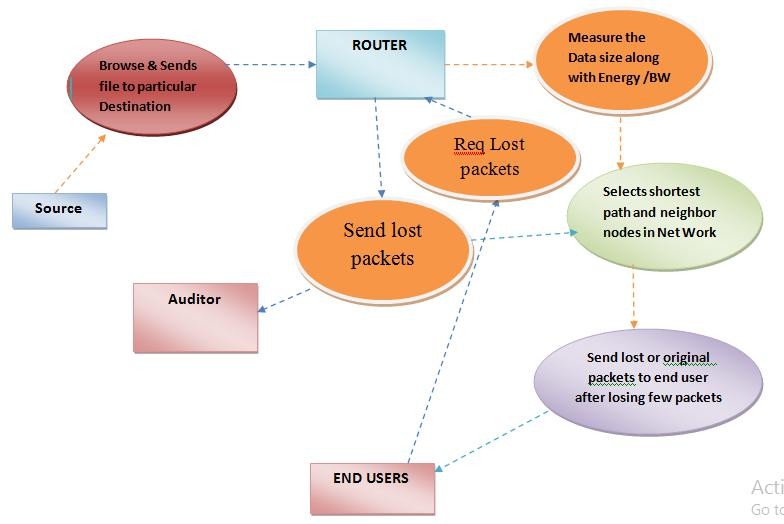
Class diagram depicts the different classes service provider, route, auditor ,attacks and end use each class consists of methods and members.

**SEQUENCE DIAGRAM**



Routing of file and shown in the diagram by sequence of activities cloud router checks attacker details and route files to receiver.

**DATA FLOW DIAGRAM**



Data flow diagrams shows specific work of each class involved in routing

**CHAPTER-8**

**CONCLUSIONS**

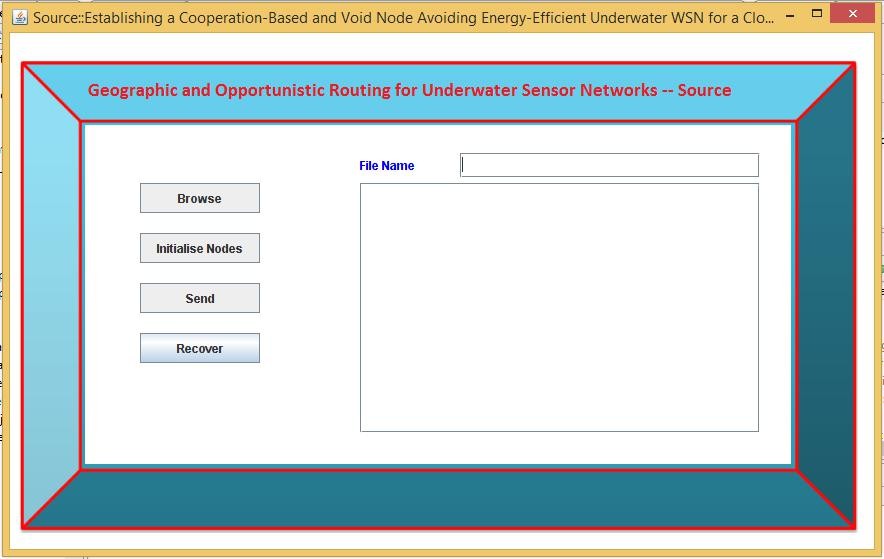
In this work, we have proposed two new schemes for underwater wireless sensor networks (UWSNs): avoiding void node adaptive hop-by-hop vector based forwarding (AVN-AHH-VBF) and cooperation based CoAVN-AHH- VBF (CoAVN-AHH-VBF). More specifically, this paper contributed in three aspects: energy efficient forwarder selection while avoiding void regions in the network, proper holding time calculation, and bit error rate BER minimization. Our forwarder selection technique resulted in high delivery ratio (DR) even in sparse network conditions. The holding time is minimized per successful packet by using our formulated. Simulation results show that both schemes are efficient in terms of energy consumption cost per dropped packet, delay and DR. Though, the energy expenditure of CoAVN-AHH-VBF is relative per received packet, this has been compensated by saving energy on every dropped packet.

The proposed non-cooperative scheme is relatively efficient in terms of delay and energy expenditure, and the proposed cooperative scheme is relatively efficient in terms of energy expenditure and DR.

# CHAPTER-9

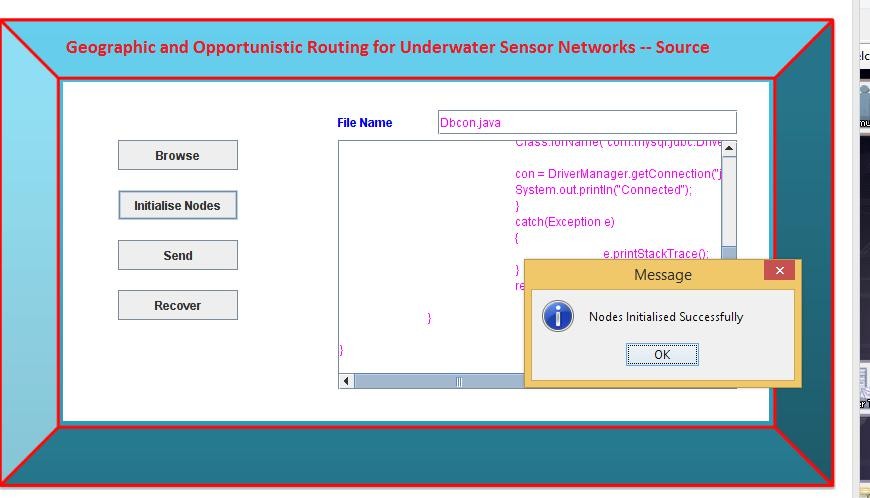
## APPENDICES APPENDIX 1:SCREENSHOTS

1. **SOURCE**

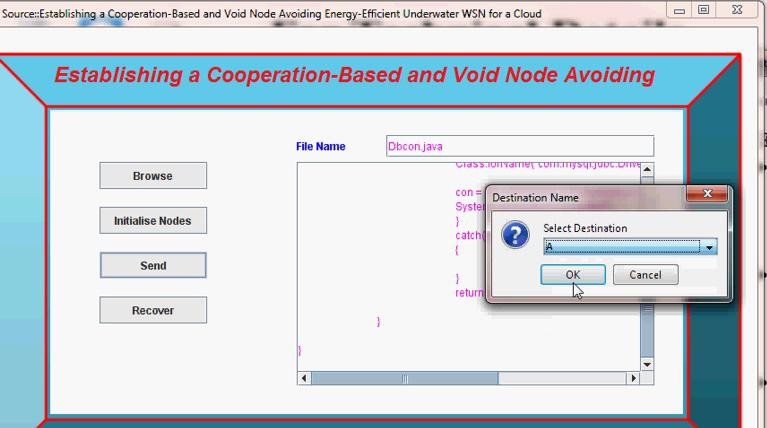


**FIG.A**

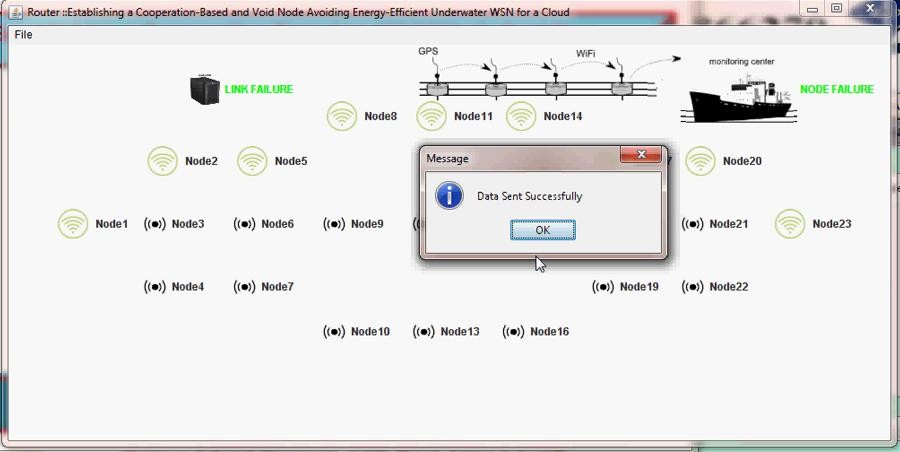
**STEP1:**It Illustrates, browsing the data and initializaiton of nodes.



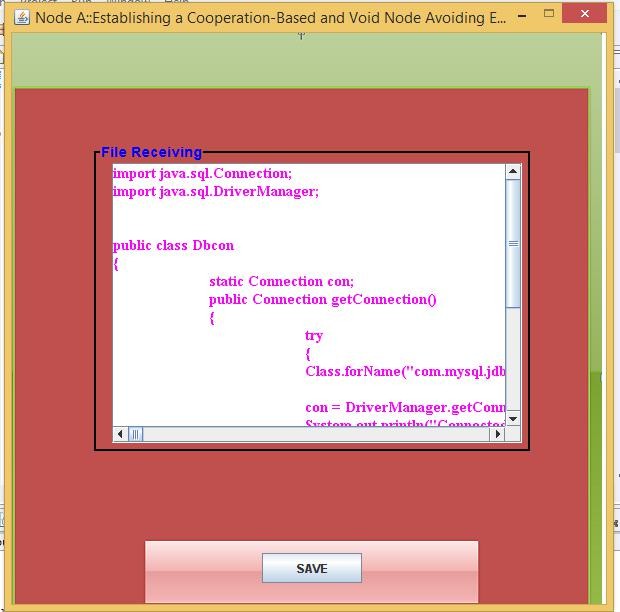
### FIG.A

**STEP2:**The destination node is selected.

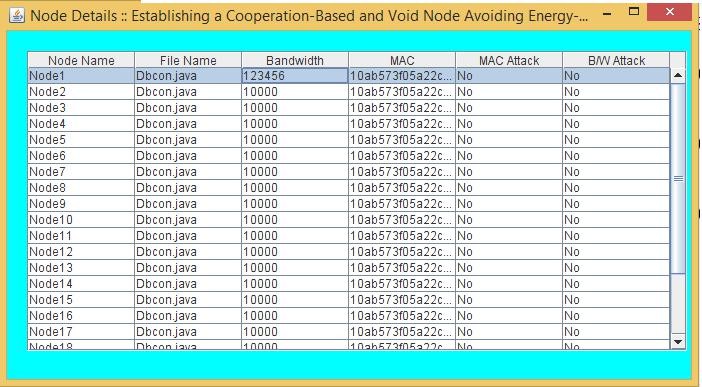
**FIG.B**Denotes the data flow to the destination node



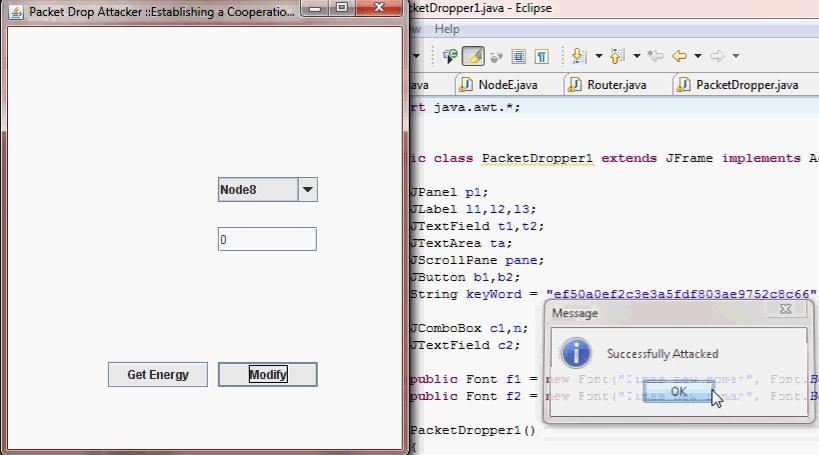
**FIG.E** Node A receiving the data.

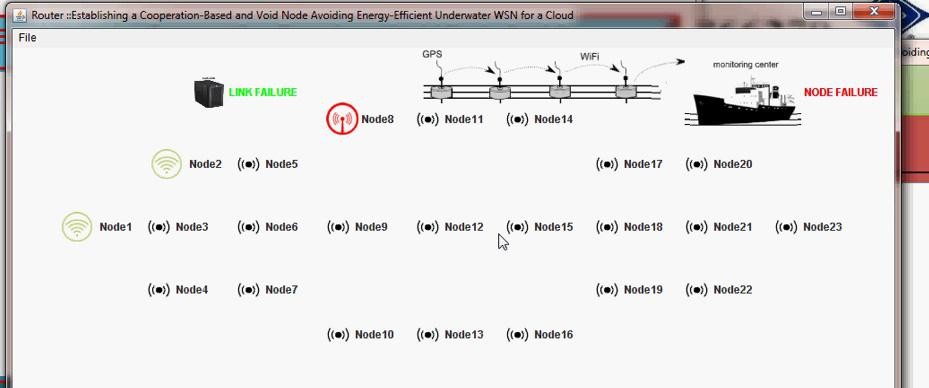


### ROUTER DETAILS:

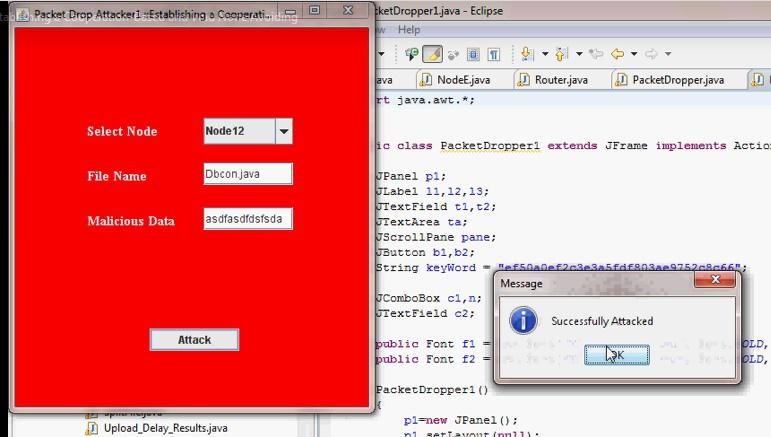


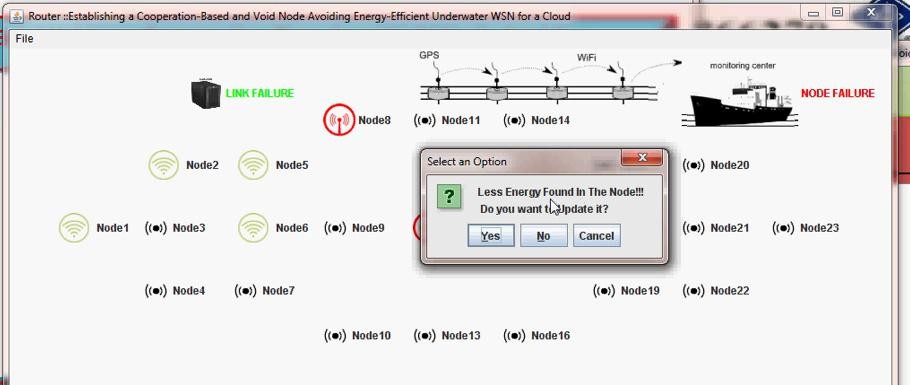
**FIG.C** Node 8 is attacked.

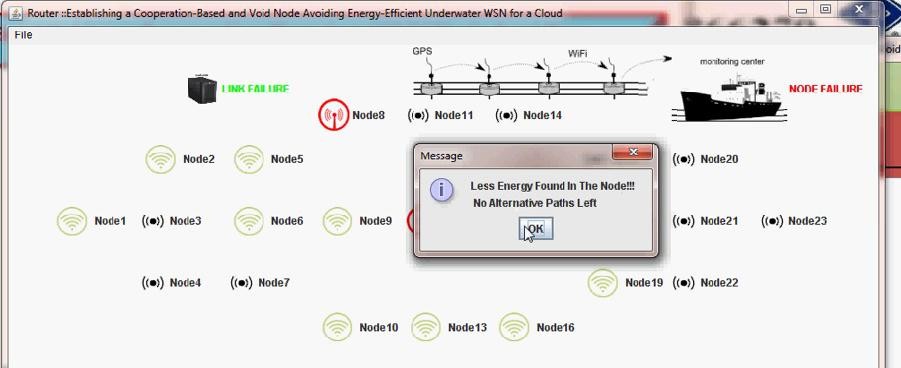




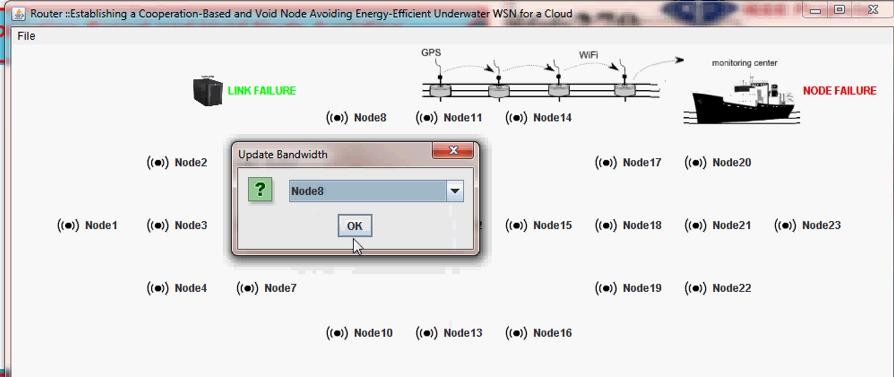
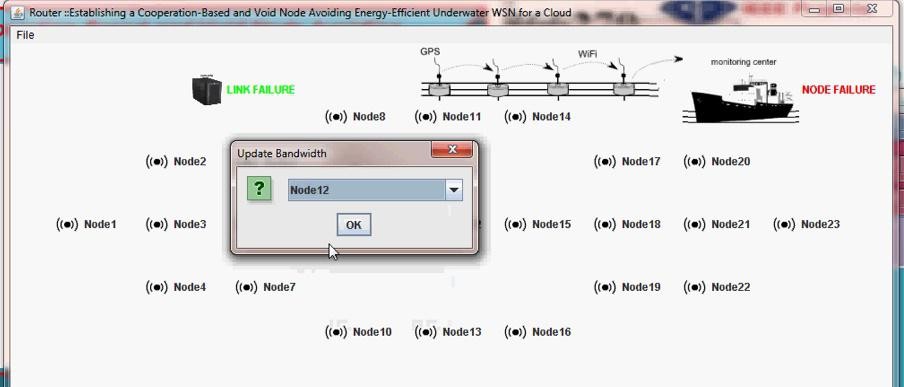
**FIG.D** Node 12 is attacked







Bandwidth is updated in Node 8 and 12.



## APPENDIX 2:SOURCE CODE

import java.awt.BorderLayout; import java.awt.Color;

import java.awt.Container; import java.awt.Font; import java.awt.Image; import java.awt.Toolkit;

import java.awt.event.ActionEvent; import java.awt.event.ActionListener; import java.awt.event.KeyEvent; import java.awt.event.WindowAdapter; import java.awt.event.WindowEvent; import java.io.BufferedInputStream; import java.io.BufferedOutputStream; import java.io.BufferedReader;

import java.io.BufferedWriter; import java.io.DataInputStream; import java.io.DataOutputStream; import java.io.File;

import java.io.FileInputStream; import java.io.FileOutputStream; import java.io.IOException;

public class Source implements ActionListener

{

byte[] b; String dd;

String txtstr=null; String hash; String filename;

String content,path1; ImageIcon ic;

public JLabel T0= new JLabel(); public JLabel confirm=new JLabel();

public JLabel T2= new JLabel("File Name"); public JLabel T4= new JLabel();

public JButton btn = new JButton("Browse"); public JTextArea tf = new JTextArea();

// btn.setFont(f3); btn1.setBounds(420,198,70,25); txt.setBounds(260,150,180,25); txt.setForeground(Color.BLUE); pane.setBounds(350, 150, 400, 250); tf.setColumns(20); tf.setForeground(Color.MAGENTA);

MessageDigest md1 = MessageDigest.getInstance("SHA1"); FileInputStream in11 = new FileInputStream("Sender\\"+ fname); DigestInputStream dis21 = new DigestInputStream(in11, md1); BufferedInputStream bd1 = new BufferedInputStream(dis21); while (true) {

int b2 = bd1.read(); if (b2 == -1)

break;

}BigInteger bi21 = new BigInteger(md1.digest()); String mac1 = bi21.toString(16); System.out.println(mac1);

SplitFile sp = new SplitFile();

sp.splitFile(new File("Sender\\"+fname),5, 1024); int len = fname.length();

String fil = fname.substring(0, len-5);

MessageDigest md = MessageDigest.getInstance("SHA1"); FileInputStream fis1 = new FileInputStream("Sender\\"+fil+"1"+".txt"); byte bs1[] = new byte[fis1.available()];

fis1.read(bs1);

String split1 = new String(bs1); fis1.close();

FileInputStream fis11 = new FileInputStream("Sender\\"+fil+"1"+".txt");

DigestInputStream dis1 = new DigestInputStream(fis11, md); BufferedInputStream bis1 = new BufferedInputStream(dis1); while (true) {

int b1 = bis1.read(); if (b1 == -1)

break;

}

BigInteger bi1 = new BigInteger(md.digest()); tring spl1 = bi1.toString();

String h1 = bi1.toString(16);

FileInputStreamfis2=newFileInputStream("Sender\\"+fil+"2"+".txt");//C:\j ava\mtechproject\cloudcomputing\pra1.txt

byte bs2[] = new byte[fis2.available()]; fis2.read(bs2);

String split2 = new String(bs2);

FileInputStream fis22 = new FileInputStream("Sender\\"+fil+"2"+".txt"); DigestInputStream dis2 = new DigestInputStream(fis22, md); BufferedInputStream bis2 = new BufferedInputStream(dis2);

while (true) {

int b2 = bis2.read(); if (b2 == -1)

break;

}

REFERENCES

# CHAPTER 10

1. P. Xie, J. Cui, and L. Lao, ``VBF: Vector-based forwarding protocol for underwater sensor networks,'' in Proc. Int. Conf. Res. Netw., 2006, pp. 12161221.
2. N. Nicolaou, A. See, P. Xie, J. Cui, and D. Maggiorini, ``Improving the robustness of location-based routing for underwater sensor networks,'' in Proc. Eur. OCEANS, 2007, pp. 16.
3. A.Wahid, S. Lee, H.-J. Jeong, and D. Kim, ``Eedbr: Energy-efcient depth- based routing protocol for underwater wireless sensor networks,'' in Proc. Adv. Comput. Sci. Informat. Technol., Sep. 2011, pp. 223234.
4. J. Khan and H. Cho, ‘‘A distributed data-gathering protocol using AUV in underwater sensor networks,’’ Sensors, vol. 15, no. 8, pp. 19331–19350, Sep. 2015.
5. N. Javaid et al., ‘‘Delay-sensitive routing schemes for underwater acous- tic sensor networks,’’ Int. J. Distrib. Sensor Netw., vol. 11, Mar. 2015, Art. no. 532676.
6. N. Sun, G. Han, T. Wu, J. Jiang, and L. Shu, ``A reliable and energy

effcient VBF-improved cross-layer protocol for underwater acoustic sensor network,'' in Proc. 11th Int. Conf. Heterogeneous Netw. Quality Rel.,

Sep. 2015, pp. 4449.

1. H. Yu, N. Yao, and J. Liu, ``An adaptive routing protocol in underwater

sparse acoustic sensor networks,'' Ad Hoc Netw., vol. 34, pp. 121143,Nov. 2015.

1. F. Mansourkiaie and M. H. Ahmed, ‘‘Optimal and near-optimal coop- erative routing and power allocation for collision minimization in wire- less sensor networks,’’ IEEE Sensors J., vol. 16, no. 5, pp. 1398–1411, Mar. 2016.
2. M. Stojanovic, ‘‘On the relationship between capacity and distance in an underwater acoustic communication channel,’’ ACM SIGMOBILE Mobile Comput. Commun. Rev., vol. 11, no. 4, pp. 34–43, 2007.

[10]K. El-Darymli, ‘‘Amplify-and-forward cooperative relaying for a linear wireless sensor network,’’ in Proc. IEEE Int. Conf. Syst. Man (SMC), Oct. 2010, pp. 106–112.