STATUS OF THESIS

Title of thesis: AN EFFICIENT COMBINED CONGESTION HANDLING AND ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC ENVIRONMENT IN BLUETOOTH NETWORK

I SABEEN TAHIR

Hereby allow my thesis to be placed at the Information Resource Center (IRC) of Universiti Teknologi PETRONAS (UTP) with the following conditions:

1. The thesis becomes the property of UTP
2. The IRC of UTP may make copies of the thesis for academic purposes only.
3. This thesis is classified as

[ ] Confidential

[√] Non-confidential

If this thesis is confidential, please state the reason:

____

____

The contents of the thesis will remain confidential for __ years.

Remarks on disclosure:

____

____

Endorsed by

Sabeen	
Signature of Author	Signature of Supervisor
Permanent address:	Associate Professor. Dr. Abas Md. Said
Makki Street, Mandian, Abbottabad, Pakistan	
Date : 22/7/2013	Date : 22/7/13

AN EFFICIENT COMBINED CONGESTION HANDLING AND ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC ENVIRONMENT IN BLUETOOTH NETWORK

by

SABEEN TAHIR

The undersigned certify that they have read, and recommend to the Postgraduate Studies Programme for acceptance this thesis for the fullfilment of the requirements for the degree stated.

Signature:	
Main Supervisor:	Associate Professor. Dr. Abas Md. Said
Signature:	
Head of Department:	Dr. Jafreezal Bin Jaafar
Date:	22/7/13

Dr. Jafreezal Bin Jaafar
Head
Department of Computer & Information Sciences
Universiti Teknologi PETRONAS
31750 Tronoh
Perak Darul Ridzuan, MALAYSIA

AN EFFICIENT COMBINED CONGESTION HANDLING AND ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC ENVIRONMENT IN BLUETOOTH NETWORK

By

SABEEN TAHIR

A Thesis

Submitted to the Postgraduate Studies Program

as a requirement for the degree of

DOCTOR OF PHILOSOPHY

INFORMATION TECHNOLOGY

UNIVERSITI TEKNOLOGI PETRONAS

BANDAR SERI ISKANDAR

PERAK, MALAYSIA

JULY 2013

DECLARATION OF THESIS

Title of thesis	AN EFFICIENT COMBINED CONGESTION HANDLING AND ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC ENVIRONMENT IN BLUETOOTH NETWORK

I, SABEEN TAHIR

hereby declare that the thesis is based on my original work except for quotations and citations, which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTP or other institutions.

Witnessed by

Sabeen

Signature of Author

Permanent address:

Makki Street, Mandian, Abbottabad, Pakistan

Date: 22/7/2013

Signature of Supervisor

Associate Professor. Dr. Abas Md. Said

Date: 22/7/13

DEDICATION

To my beloved family, specially to my mother (Maryam Rauf)

ACKNOWLEDGEMENT

All praise is due to Allah (SWT), who formed man and qualified him what he knew not. I am thankful to Allah (SWT) for enhancing my spirit for the successful accomplishment of this work. Allah helped me in doing a lot of research, and I came to learn, diligently, about so many new things.

Secondly, I would like to express my special gratitude to my respectable teacher, Associate Professor Dr. Abas Md. Said; he gave me moral support and guided me in different matters regarding my research topic.

I would also like to thank my colleagues, who often helped and gave me support at critical junctures during this research. I also appreciate the efforts and cooperation of the management of Universiti Teknologi PETRONAS, Malaysia, for their assistance and cooperation in this research work.

Last but not the least, I would like to express thank my family who helped me a lot and guided me from time to time in completing this research. There are no words which can convey my appreciation for Dr. Tahir and Eng. Aman despite their busy schedule; they gave me different ideas to help make this research unique. Lastly, I apologize if I have irritated anybody for the duration of my studies in this remarkable institution.

ABSTRACT

Bluetooth is a widespread technology for small wireless networks that permits Bluetooth devices to construct a multi-hop network called a scatternet. A large number of connections passing through a single master/ bridge device may create the problem of congestion in a Bluetooth scatternet. In addition, routing in a multi-hop dynamic Bluetooth network, where a number of masters and bridges exist, sometimes creates technical hitches in a scatternet. It has been observed that frequent link disconnections and a new route construction consume more system resources that ultimately degrade the performance of the whole network. As, Bluetooth specification has defined piconet configuration, scatternet configuration has still not been standardized. The main objective of this thesis is to provide an efficient combined protocol for scatternet congestion handling and route maintenance. The methodology contains three parts.

The first part consists of an Efficient Congestion Handling Protocol for Bluetooth Scatternet (ECHP) that solves the problem of Bluetooth network congestion. ECHP is further divided into two sub-parts, the first sub-part is based on intra-piconet congestion handling, while the second sub-part is based on inter-piconet congestion handling. In the second part of this research work, an Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks (ERMP) is proposed that repairs the weak routing links based on the prediction of weak links and weak devices. The ERMP has two constraints of prediction; the first constraint is mobility and the second is energy based. The ERMP predicts a weak link through the signal strength and weak devices through the energy level of the devices. During the main route construction, routing masters and bridges keep the information of the Fall Back Devices (FBDs) for route maintenance. On the prediction of a weak link, the ERMP activates an alternate link, and on the prediction of the weak device it activates the FBD before the main link breaks. In the third part of the thesis, An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network

(CHRM) is proposed. The CHRM has merged both of the proposed ECHP and ERMP protocols and provides an improved solution for inter-piconet communication.

For simulation, UCBT is used which is based on an open source simulator NS-2 that presents an absolute simulation model for Bluetooth performance evaluation on different Bluetooth layers. The existing closely related protocols and the proposed protocols are implemented through simulation and then compared. The simulation results reveal that the proposed protocols efficiently solved the scatternet congestion and increased throughput 25-35%. Moreover, the proposed protocol repairs the weak routing links and reducing delay 30-40%.

ABSTRAK

Bluetooth adalah sebuah teknologi yang meluas untuk rangkaian kecil tanpa wayar yang membolehkan peranti Bluetooth untuk membina satu rangkaian pelbagai-hop dipanggil scatternet. Sebilangan besar sambungan melalui sarjana / jambatan peranti tunggal boleh menyebabkan masalah kesesakan di scatternet Bluetooth. Di samping itu, laluan di rangkaian Bluetooth multi-hop yang dinamik, di mana beberapa tuan-tuan dan jambatan wujud, kadang-kadang mewujudkan hitches teknikal dalam scatternet. Ia telah diperhatikan bahawa kerap link terputus dan pembinaan laluan baru menggunakan sumber-sumber sistem yang lebih yang akhirnya merendahkan prestasi rangkaian keseluruhan. As, spesifikasi Bluetooth telah ditakrifkan konfigurasi rangkaian pico, konfigurasi scatternet masih tidak seragam. Objektif utama projek ini adalah untuk menyediakan protokol digabungkan berkesan untuk pengendalian kesesakan scatternet dan penyelenggaraan laluan. Kaedah ini mengandungi tiga bahagian.

Bahagian pertama terdiri daripada Kesesakan Cekap Protokol Pengendalian untuk Bluetooth Scatternet (ECHP) yang menyelesaikan masalah kesesakan rangkaian Bluetooth. ECHP dibahagikan kepada dua sub-bahagian, yang pertama sub-bahagian adalah berdasarkan pengendalian kesesakan antara rangkaian pico, manakala yang kedua sub-bahagian adalah berdasarkan pengendalian kesesakan antara rangkaian pico. Dalam bahagian kedua kerja-kerja penyelidikan ini, Route Cekap Protocol Penyenggaraan Rangkaian Bluetooth Dinamik (ERMP) adalah dicadangkan pembaikan pautan laluan lemah berdasarkan ramalan hubungan yang lemah dan peranti yang lemah. ERMP mempunyai dua kekangan ramalan; kekangan yang pertama adalah mobiliti dan yang kedua adalah berdasarkan tenaga. ERMP meramalkan pautan yang lemah melalui kekuatan isyarat yang lemah dan peranti melalui tahap tenaga alat-alat. Semasa pembinaan laluan utama, laluan tuan dan jambatan menyimpan maklumat daripada Devices Kembali Kejatuhan (FBDs) bagi

penyelenggaraan laluan. Pada ramalan link yang lemah, ERMP mengaktifkan link alternatif, dan pada ramalan peranti yang lemah ia akan mengaktifkan FBD sebelum rehat pautan utama. Dalam bahagian ketiga tesis, Satu Gabungan Cekap Pengendalian Kesesakan dan Route Protocol Penyelenggaraan Alam Sekitar Dinamik dalam Bluetooth Network (CHRM) dicadangkan. CHRM telah digabungkan kedua-dua ECHP cadangan dan protokol ERMP dan menyediakan penyelesaian yang lebih baik untuk komunikasi antara rangkaian pico.

Untuk simulasi, UCBT digunakan yang berasaskan sumber terbuka simulator NS-2 yang membentangkan model simulasi mutlak untuk penilaian prestasi Bluetooth Bluetooth lapisan yang berbeza. Protokol yang sedia ada berkait rapat dan protokol yang dicadangkan adalah dilaksanakan melalui simulasi dan kemudian dibandingkan. Keputusan simulasi menunjukkan bahawa protokol yang dicadangkan cekap menyelesaikan kesesakan scatternet dan peningkatan pemprosesan 25-35%. Selain itu, pembaikan protokol cadangan yang lemah laluan hubungan dan mengurangkan kelewatan 30-40%.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

ACL	Asynchronous Connection Less
AM_Addr	Active Member Address
AR_Addr	Access Request Address
ARQN	Automatic Repeat Request Number
AxM	Auxiliary Master
AxB	Auxiliary Bridge
BB	Base Band
BD_Addr	Bluetooth Device Address
BR	Backup Relay
BTFT	Bridge Traffic Flow Table
CAC	Channel Access Code
CLKN	Bluetooth Native Clock
DAC	Device Access Code
DCC	Dynamic Congestion Control
ECHP	Efficient Congestion Handling Protocol
DIAC	Dedicated Inquiry Access Code
DCI	Default Check Initialization
ERMP	Efficient Route Maintenance Protocol
FBB	Fall Back Bridge
FBD	Fall Back Devices
FHS	Frequency Hopping Synchronization
FHSS	Frequency Hopping Spread Spectrum
GIAC	General Inquiry Access Code
HCI	Host Control Interface

HEC	Header-Error-Check
ID	Node ID
IAC	Inquiry Access Code
ISM	Industrial, Scientific and Medical
LAP	Lower Address Part
LMP	Link Manager Protocol
L2CAP	Logical Link Controller and Adaptive Protocol
PRP	Piconet Restructuring Protocol
MANET	Mobile Ad Hoc Network
MFN	Most Frequently used Nodes
MIT	Master Information Table
MSM	Master/Slave Mesh
MSR	Master/Slave Ring
MTFT	Master Traffic Flow Table
NAP	Non-significant Address Part
NIT	Node Information Table
NS	Network Simulator
OBEX	Object Exchange Protocol
PAN	Personal Area Network
PDAs	Personal Digital Assistants
PDU	Bluetooth Protocol Data Unit
PM	Park Mode
PFP	Piconet Formation within Piconet
PM_Addr	Parked Member Address
PPP	Point-to-Point Protocol
QoS	Quality of Service

RBIT	Routing Bridge Information Table
RF	Radio Frequency
RMIT	Routing Master Information Table
ROMA	Novel Route Maintenance
RR	Route Request
RRP	Route Reply Packet
RSP	Route Search Packet
Rx	Receiver
SCO	Synchronous Connection Oriented
SDP	Service Discovery Protocol
SFP	Scatternet Formation within Piconet
SIG	Special Interest Group
SNR	Signal-to-Noise Ratio
TDD	Time Division Duplex
TFT	Traffic Flow Table
TDF	Total Data Flow
Tx	Transmitter
UAP	Upper Address Part
UART	Universal Asynchronous Receiver/Transmitter
UCBT	University of Cincinnati Bluetooth
WAP	Wireless Application Protocol
WPAN	Wireless Personal Area Network

CHAPTER 1

INTRODUCTION

This chapter covers the background, motivation, the critical research issues of existing Bluetooth scatternet protocols based on which this research work has been carried out. It also includes the objectives to be achieved and illustrates the research questions and their answers. The chapter continues the proposed research methodology that describes, an efficient congestion handling protocol for Bluetooth scatternet, an efficient route maintenance protocol for dynamic Bluetooth networks and a new combined protocol of congestion control and route maintenance for a dynamic Bluetooth scatternet.

1.1 BACKGROUND

The improvement in the capability of mobile devices has enhanced the usage of mobile digital devices like computer systems, cellular phones and Personal Digital Assistants (PDAs). These kinds of devices increase the usage of wireless communication technology every day. Wireless communication technologies can be classified into many factors. One of them is the communication range; mobile devices for example mobile computers or PDAs, usually require a short range for wireless access because these devices normally operate with limited battery power. These wireless devices also have high bandwidth capacity. This type of short range, low power and high data rate wireless technology is called a Wireless Personal Area Network (WPAN) (Movahhedinia and Ghahfarokhi 2011) that supports the operation of an ad hoc mode and the mobility of devices. Bluetooth technology is one of the short range wireless technologies. This technology has attracted the particular attention of users and vendors (Chen and Liu 2006; Kazemian and Li 2006; Shek and Kwok 2004).

The Bluetooth Special Interest Group (SIG) was established in 1998 (Ka-Wai and Luong 2003; Lee and Lee 2010). Initially, it consisted of five big companies: IBM, Ericsson, Intel, Nokia, and Toshiba but now more than 16000 companies have joined SIG to improve the idea of the Bluetooth open standard (Jong Min et al. 2005). The main objective of Bluetooth technology is cable replacement and connection of the devices wirelessly within a short range in an ad-hoc manner. Bluetooth technology can be used in many applications which are categorized into several groups, such as, workplace applications, ad-hoc networking and access control applications (Chaudhry et al. 2008; Kiss Kalló et al. 2007; McDermott-Wells 2004).

Workplace applications: These applications are associated with the setting up of a wireless workplace environment. For example, by using a Bluetooth link, a mouse and a keyboard are connected with a computer, a Bluetooth enabled printer can connect many computers at a time and a Bluetooth technology can be used for the data synchronization between the Bluetooth enabled devices (Khan et al. 2009).

Ad-hoc networking: Bluetooth enabled devices can make a large ad-hoc network commonly named the scatternet. As the devices in ad-hoc networks can connect and exist as long as required if they are within the radio range of each other. Currently, these applications are used to exchange the electronic business cards between users(Chaudhry et al. 2008).

Access control applications: Bluetooth technology can be used in an access card for the purpose of having the right of entry in a building or accessing resources without using a swipe machine. This technology can be used for mobile payment (credit card) (Yunbo et al. 2007).

1.2 MOTIVATION

The Bluetooth fundamental system consists of a Radio Frequency (RF) transceiver, protocol stack and Baseband (Chen et al. 2009). The Bluetooth RF function is internationally accessible with a 2.4 GHz Industrial, Scientific and Medical (ISM) band where hopping covers 79 channels with 1MHz bandwidth. The time interval between different frequencies is $625\mu s$. The Frequency Hopping Spread Spectrum (FHSS) system is used to decrease the interference and increase the security in the Bluetooth network (Khan et al. 2009; Xin and YuPing 2009). The Bluetooth connection establishment procedure is controlled by four processes: 1) Inquiry: In this process a Bluetooth device broadcasts inquiry packets to find the IDs of other Bluetooth devices. 2) Inquiry scan: In the inquiry scan process, other Bluetooth devices listen for inquiry packets and broadcast inquiry response packets. 3) Page: Under this procedure, any Bluetooth device attempts to make a link with the device whose ID and clock are recognized. 4) Page scan: In the page scan process, Bluetooth devices listen for page packets. The basic communication unit of Bluetooth is called piconet which consists of a maximum of eight active Bluetooth nodes/devices. Bluetooth allows communication within multiple piconets where a relay or bridge device provides connections among different piconets. A group of multiple, connected piconets are called a scatternet (Kapoor et al. 2004; Mišić and Mišić 2004; Vergetis et al. 2005).

The motivation of this thesis is to demonstrate that the efficient formation of a scatternet and routing techniques can easily handle the Bluetooth network congestion and route maintenance issues. Only few protocols have been proposed for network congestion handling and for route maintenance but they still have some problems due to neglecting some main issues that leads the inefficient scatternet formation. The existing protocols create more delay; increases control packets overhead and decrease the network throughput. The existing route maintenance protocols have been implemented to solve the routing issues after the main link failure. Therefore, an efficient congestion handling and route maintenance protocols have been proposed in this thesis which is based on the prediction of the signal strength and energy level of devices. The proposed protocols provide the solution for congestion handling and route maintenance before the main link breakdown. As a result, it reduces the control overhead, minimizes the time required for route recovery and it improves the network throughput.

1.3 PROBLEM STATEMENT

A lot of scatternet optimization techniques have been developed but each technique has its own benefits and limitations. It is analyzed that a combined (congestion and route maintenance) solution for scatternet has been neglected. Thus, there is still need for an efficient and optimized scatternet formation. Many researchers have proposed scatternet formation protocols to decrease the scatternet formation time or increase the probability of making a scatternet. The main purpose of this thesis is to propose a combined protocol for congestion handling and scatternet route maintenance.

If Bluetooth devices communicate within a piconet, then all communicating links go through the master device, and if Bluetooth devices from other piconets communicate with each other, then all communicating links go through the intermediate device called a bridge/relay device. When a large number of connections pass through a single master or bridge device, it may cause congestion and delay in the Bluetooth scatternet. Within a piconet, direct slave to slave communication is not possible, all outgoing and incoming data traffic go through the master device so it is considered as a very important device. If a master device drains out its battery or moves, it will disconnect all of the linked devices. Similarly, within a scatternet, direct master to master communication is not possible so they always need an intermediate bridge device therefore, the bridge device is considered as the most significant device because it connects multiple piconets. If it fails or moves, then it can disturb the whole network. When the data is routed in a Bluetooth network for slave to slave communication, it follows the rule that the data will go through "slave-to-master- bridge-master-slave".

In a piconet, any couple of source and destination is at most two hops away. A master device maintains the information about all of its slave devices and controls the incoming and outgoing data traffic. Whereas, in a scatternet, a couple of source and destination devices may be in different piconets so the data packets can go through multiple hops. A considerable amount of work has been done in the area of Bluetooth network routing. The most important Bluetooth inter-piconet communication issue is route maintenance for broken links which has not been considered comprehensively. Some Bluetooth routing protocols provide a solution for route recovery when they predict the link breakage. The main disadvantages with these types of protocols are they are more time consuming and cause extra resource consumption.

1.4 OBJECTIVES

To achieve an efficient communication within a well organized Bluetooth network, the following objectives are defined.

1.4.1 Bluetooth network congestion handling

The first objective of this research is to illuminate the problem of congestion in a Bluetooth network. This is because a large number of connections passing through a single master/bridge node may create the congestion problem in a Bluetooth scatternet. To overcome the problem of congestion, An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP) is proposed in this thesis. The proposed protocol is based on the Fall Back Device (FBD) utilization and the operation of the network restructuring. The proposed ECHP protocol has two parts; the first part demonstrates how the congestion in an intra-piconet is handled. Whereas, the second part demonstrates how the inter-piconet congestion can be handled.

1.4.2 Bluetooth network route maintenance

The second objective of this research work is to achieve the efficient route maintenance in a scatternet. As a main routing link in a multi-hop dynamic Bluetooth network has a number of master and bridge devices, this sometime creates technical hitches (link breakage etc.) that consume extra resources and ultimately degrade the performance of the whole network. In this thesis, an Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks (ERMP) is proposed that repairs the weak routing links based on the prediction of weak links and weak devices. It predicts the weak links and weak devices by determining the weak signal strength between the devices and the weak devices having the minimum energy level. During the main route construction, the routing master and routing bridge devices keep the information of the FBD for route maintenance. On the prediction of a weak link, they activate an alternate link. While, on the prediction of a weak device, they activate FBD before the main link breakage.

1.4.3 An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network (CHRM)

The third objective of this research work is to implement a new protocol "An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network (CHRM)" that combines both of the proposed protocols (ECHP and ERMP). In this protocol, the problems of both of the protocols are considered and a combined solution is proposed. The simulation results are achieved by comparing the combined protocol CHRM with the previous DCC and ROMA protocols.

1.5 RESEARCH QUESTIONS

The following are the research questions which are based on the mentioned research issues.

a. How can the resources be utilised efficiently?

   As the Bluetooth technology has limited resources, therefore, the resources should be utilized efficiently. Resources can be used efficiently if the scatternet formation is efficient. It should not contain the maximum bridge devices or extra piconets, and it does not waste the Active Member Address (AM_Addr) because the inefficient utilization of AM_Addr creates the problem of data packet loss and delay.

b. How can Bluetooth network congestion be handled?

Mostly, the problem of congestion arises on the master or bridge devices because of providing a large number of routing links. Some protocols have been proposed for congestion handling but they could not provide an efficient solution. They cannot allow the parallel transmissions either. The proposed protocol handles the problem of congestion by using the Fall Back Bridge (FBB) based on the role switch operations. The higher throughput of the proposed protocol shows that it delivers the maximum data packets by a device per unit time.

c. How can the time required for route repairing in route maintenance be reduced?

During transmission, if the main route breaks, then the route re-establishment procedure takes more time for route repairing. All of the previous protocols proposed for Bluetooth network route repairing are based on the route re-establishment procedure or the prediction of the energy level. They totally ignore some important constraints and as a result they create more delay time. As the proposed protocol provides the route repairing solution on run time by the prediction of weak devices, therefore, it does not require more time for route repairing. It updates the list of FBDs during the main route construction.

d. How can control overhead be reduced in a Bluetooth network?

In a Bluetooth network, control overhead can be reduced if the scatternet formation avoids the frequent creation of extra piconets. Bluetooth technology utilizes the maximum number of control packets during the connection establishment procedure.

1.6 PROPOSED METHODOLOGY

The proposed methodology of this research work is composed of the following main parts:

1.6.1 An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP)

An efficient congestion handling protocol for Bluetooth scatternet is proposed for an efficient Bluetooth network. This protocol handles the problem of congestion within intra-piconet and inter-piconet. This protocol consists of two scenarios; in the first scenario, role switching techniques are used to handle the problem of intra-piconet congestion which has two cases: Piconet Formation within Piconet (PFP) and Scatternet Formation within Piconet (SFP). Network restructuring is performed for the most frequently communicating pair of devices. In the second scenario, a Fall-Back Bridge (FBB) is used to share the traffic load on the bridge node that overcomes the problem of congestion in the inter-piconet.

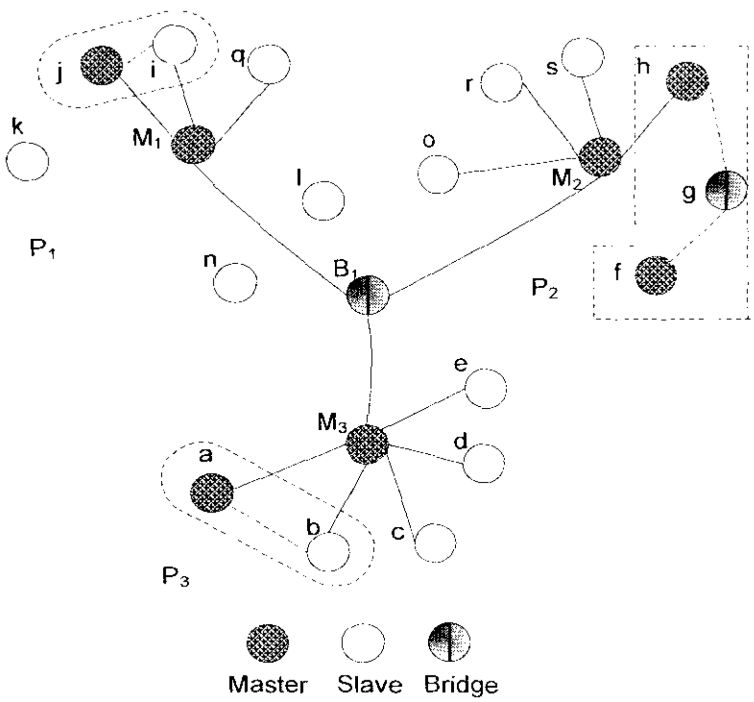


Figure 1.1: Congestion handling by making PFP and SFP using the role switch operation

The proposed protocol is explained by an example in Fig.1.1, it is depicted that there are three piconets in a scatternet. Different devices in different piconets communicate through their master devices. After some time, the master devices (M_1, M_2, M_3) analyze from their tables that in piconet P_1 devices (i and j), in Piconet P_2 devices (h and f) and in piconet P_3 devices (a and b) are communicating very frequently and creating congestion on their respective master devices. Therefore, to handle the congestion, the master devices perform network restructuring by role switch operations. The master devices ($M_1$and M_3) perform Piconet Formation within Piconet (PFP) by changing the role of the devices. Now, device (j) is the auxiliary master (AxM, a slave device that has potential to become a master device) for device (i) and device (a) is the AxM for device (b). In piconet P_2, devices (h and f) are communicating very frequently but these are not within the direct radio range of each other so M_2 has selected the device (g) as an intermediate device and performed SFP. In the new scatternet, devices (h and f) are auxiliary masters and device (g) is a bridge device. Now, the devices are communicating directly and after a successful communication the devices will return in their original states. The proposed protocol handles the inter piconet congestion by activating the FBB device. The foremost advantage of using FBB is that it also allows for parallel transmissions between multiple piconets.

1.6.2 An Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks (ERMP)

An Efficient Route Maintenance Protocol (ERMP) for Bluetooth networks is proposed. Initially, all devices in Bluetooth scatternet create required links and start communication. As routing links pass through master(s) and bridge(s) devices therefore, after the main route construction, each routing master and routing bridge device keeps the information of all connected devices in their tables. They monitor the signal strength and energy level of all of the devices. The proposed protocol predicts the weak links and weak devices through the signal strength and energy level of the devices and repairs the weak links by activating FBDs. The ERMP provides solution for route breakage, if a weak routing link is predicted, then this protocol makes a new link before the main route breaks.

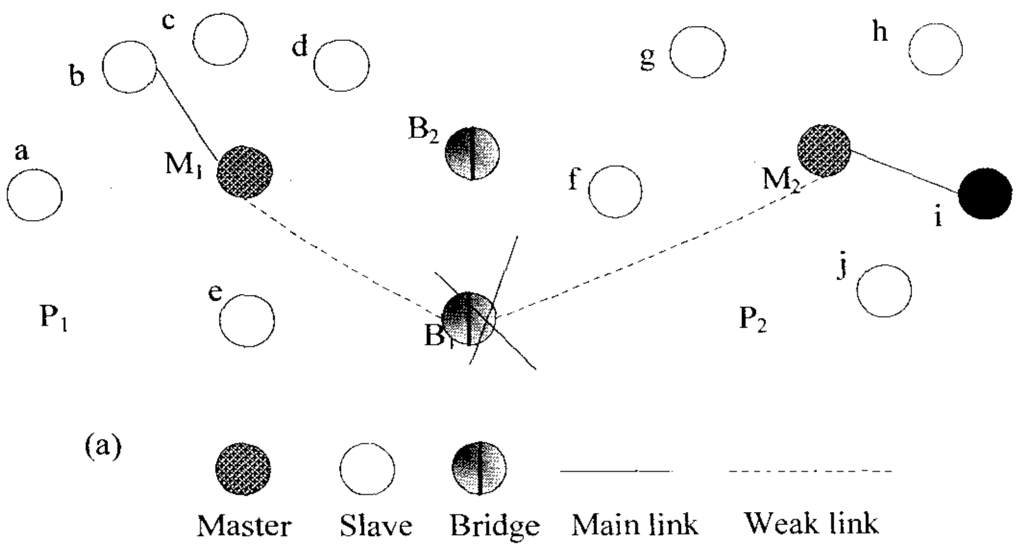


Figure 1.2(a): An example of predicting a weak link

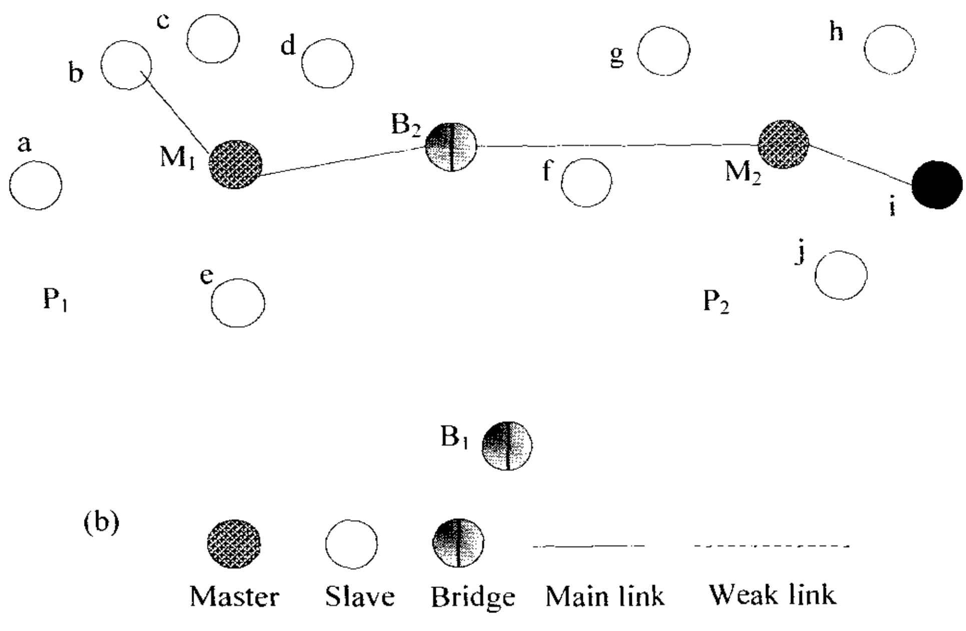


Figure 1.2(b): Route recovery by using ERMP

This protocol is explained by an example in Fig.1.2. In Fig.1.2(a) two devices (b and i) in different piconets are communicating through a main link (b-M_1-B_1-M_2 -i). During transmission between (b and i), the bridge device B_1 starts moving downward;

in the meanwhile, the routing masters M_1 and M_2 predict by the weak signal strength that the bridge device B_1 is moving so they look for the FBD as they already have the list for FBDs in their tables. Both of the routing master devices find device B_2 which is also a subset of their piconets. Both master devices transmit a request packet to device B_2 to make a new link between M_1 and M_2. This protocol has to make a new link before the main route breaks. As in Fig.1.2(b), the new link between M_1 and M_2 has been established and the data is going through the new route (b-M_1-B_2-M_2-i). Similarly, during transmission, if any weak device is predicted, then ERMP activates the FBD and makes a new link with a new device.

1.6.3 An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network (CHRM)

The third part of the proposed methodology is to make a new combined protocol CHRM that combines the ECHP and ERMP protocols. The CHRM provides a combined solution for congestion handling and route maintenance. It is observed that CHRM provides an efficient solution for Bluetooth scatternet to overcome the congestion and route breakage problems.

1.7 SCOPE

This section has outlined the scope of this dissertation. An important aspect of this research work was to assess the existing protocols used in congestion handling and route maintenance and to select some of the most related studies to proposed work. The solution proposed for load handling and route maintenance in this study helped to improve the performance of the Bluetooth scatternet. In this thesis, a Bluetooth intra-piconet and inter-piconet Congestion handling protocol has been proposed. For traffic load handling, different role switching techniques and Fall Back Devices were used to overcome the bottleneck problem. Furthermore, an efficient route maintenance protocol has been proposed which has been based on the prediction of weak devices and weak links. To evaluate the system performance, the research work and existing protocols were simulated and compared.

1.8 AN OUTLINE OF THE THESIS

The remainder of this thesis is categorized as:

Chapter 2 discusses the fundamentals of Bluetooth communication technology. This includes the connection establishment process, piconet formation, scatternet formation etc. Moreover, this chapter discusses the related work and shortcomings of existing protocols.

Chapter 3 demonstrates the methodology. In which the proposed protocols: the congestion handling protocol (ECHP), route maintenance protocol (ERMP) and a combined protocol (CHRM) have been discussed.

Chapter 4 explains the results of the proposed protocols. The proposed protocols are implemented in the University of Cincinnati's Bluetooth (UCBT) based on the NS-2 open source simulator. In this section different parameters are measured and compared with the existing protocols.

Chapter 5 presents the contribution and conclusion that describes the summary of the performance of proposed protocols and this also contains the possible future work.

A complete list of research publications throughout this research work is given at the end.

Appendix A explains the terminologies used in the thesis.

Appendix B explains the basic physical model of Bluetooth.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the fundamentals of Bluetooth communication technology are presented, which cover the connection establishment process, piconet formation and scatternet formation etc. The chapter describes the Bluetooth applications and the comparison of Bluetooth with other wireless technologies. This chapter also discusses the role switching techniques and taxonomy of the Bluetooth network topologies. Finally, the closely related protocols and shortcomings of these protocols are discussed in detail.

2.1 BLUETOOTH OVERVIEW

In these days, many people like to carry handheld devices to keep in touch with other people. So they can easily share the data (text, video and voice) with each other. Bluetooth is the most popular wireless technology to provide a connection between short-ranged devices (Le Sommer and Ben Sassi 2010; Yip and Kwok 2007). Wireless networks can be divided into two categories, infrastructure and infrastructure-less. The infrastructure networks are those networks which have immovable and wired access points like cellular network systems in which a base station is fixed that acts as a bridge to provide connections to the mobile devices. The other form of network is wireless or infrastructure-less, which is also recognized as an ad hoc network. The infrastructure less networks has no fixed access points (Cordeiro et al. 2003; Jung et al. 2008).

In 1998, a cluster of companies familiarized an open standard for squat range wireless connections named Bluetooth technology. Bluetooth technology is a promising wireless technology that permits the moveable devices to make a short ranged network (Al-Jarrah and Megdadi 2009). Its specification is specially designed for the low power two-way radio communication to wirelessly connect the network devices over short distances (Blom et al. 2008). Bluetooth devices are unable to transfer data to each other until they create a Bluetooth network. If two Bluetooth devices want to communicate with each other, they first make a small network known as a piconet in which one device will become a master and the other devices will be its slaves. In the piconet, a maximum of eight Bluetooth devices connect with each other by sharing a common channel (Mišić and Mišić 2004; Ramana Reddy et al. 2010). The piconet supports point-to-point (master to slave) and point-to-multi point (master to slaves) links as in Fig.2.1 (Shek and Kwok 2004).

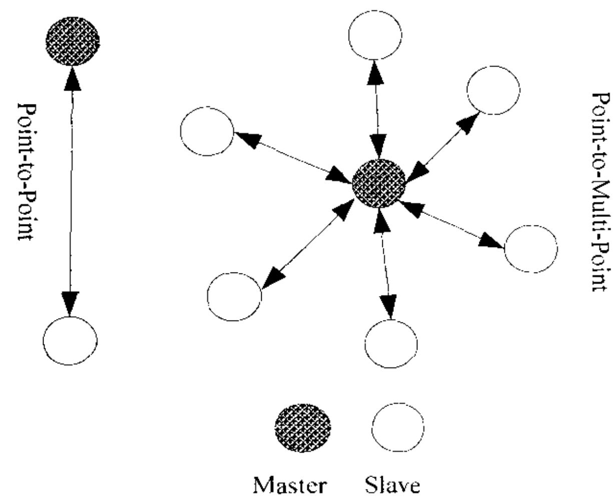


Figure.2.1: Bluetooth Piconet topologies

Bluetooth technology works on the physical layer that uses the 2.4 GHz ISM frequency band. The key advantage of the ISM frequency band is that it makes the Bluetooth technology accepted worldwide (Ophir et al. 2004; Salonidis et al. 2005). Each Bluetooth device uses 79 channels using a pseudo-random hopping sequence. The frequency (f) of a Bluetooth device for a given time slot (t) can be calculated by the equation 2.1:

$$f = 2402\text{MHz} + f(t) \times 1\text{MHz} \qquad \text{where } f(t) = 0, 1, 2 \ldots 78. \tag{2.1}$$

Each Bluetooth packet is transmitted at a different frequency. Bluetooth slave devices cannot communicate directly; they always send their data to the master device. The master device always assigns different time slots with different frequencies to the slave devices. Bluetooth equipment uses the Time Division Duplex

(TDD) scheme that breaks up a channel into 625 μs time slots. During transmission, the data packets take up 1, 3 or 5 time slots and according to the frequency hopping sequence, all of the data packets are transmitted on different physical channels (Cordeiro et al. 2003; Lee and Lee 2010; Naik et al. 2008). For a packet that takes multiple time-slots for transmission, the frequency is not changed until the whole packet is transmitted. Fig.2.2 shows the example of multi-slot packets between master and slave devices.

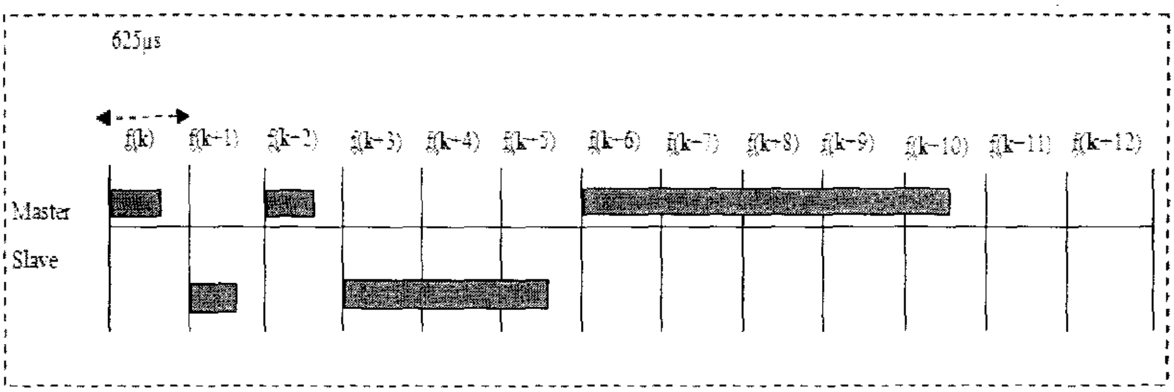


Figure 2.2: Multi-slot packets

A master device always transmits its data in even numbered time slots ($CLK_1 = 0$) and the slave device always transmits its data in odd numbered time slots ($CLK_1 = 1$). For some packet types that cover up more time slots, the transmission of the master device can be prolonged in odd numbered slots and similarly, the transmission of the slave device can be prolonged in even numbered slots. As each Bluetooth device uses a running clock, the slave devices stay synchronized to the piconet from time to time adding a timing offset to their clocks from the clock of the master. All Bluetooth devices have a unique and permanent 48-bit Bluetooth Device Address (BD_Addr). It is converted into three parts; the first part is a 24-bit Lower Address Part (LAP), the second part is an 8-bit Upper Address Part (UAP) and the third part consists of a 16-bit Non-significant Address Part (NAP). The UAP and NAP identify the device's manufacturing company, whereas the LAP is a value assigned by the company (Jung et al. 2008; Moron et al. 2009; Shek and Kwok 2004). In a Bluetooth network, each active slave device is also allocated a 3-bit AM_Addr by the master device.

2.2 BLUETOOTH PROTOCOL STACK

The Bluetooth technology consists of a layered protocol architecture that completes the core part of the Bluetooth implementation. It helps the Bluetooth devices to find each other and setup a connection. This connection enables the devices to exchange different kind of data. Fig. 2.3 shows the Bluetooth protocol stack.

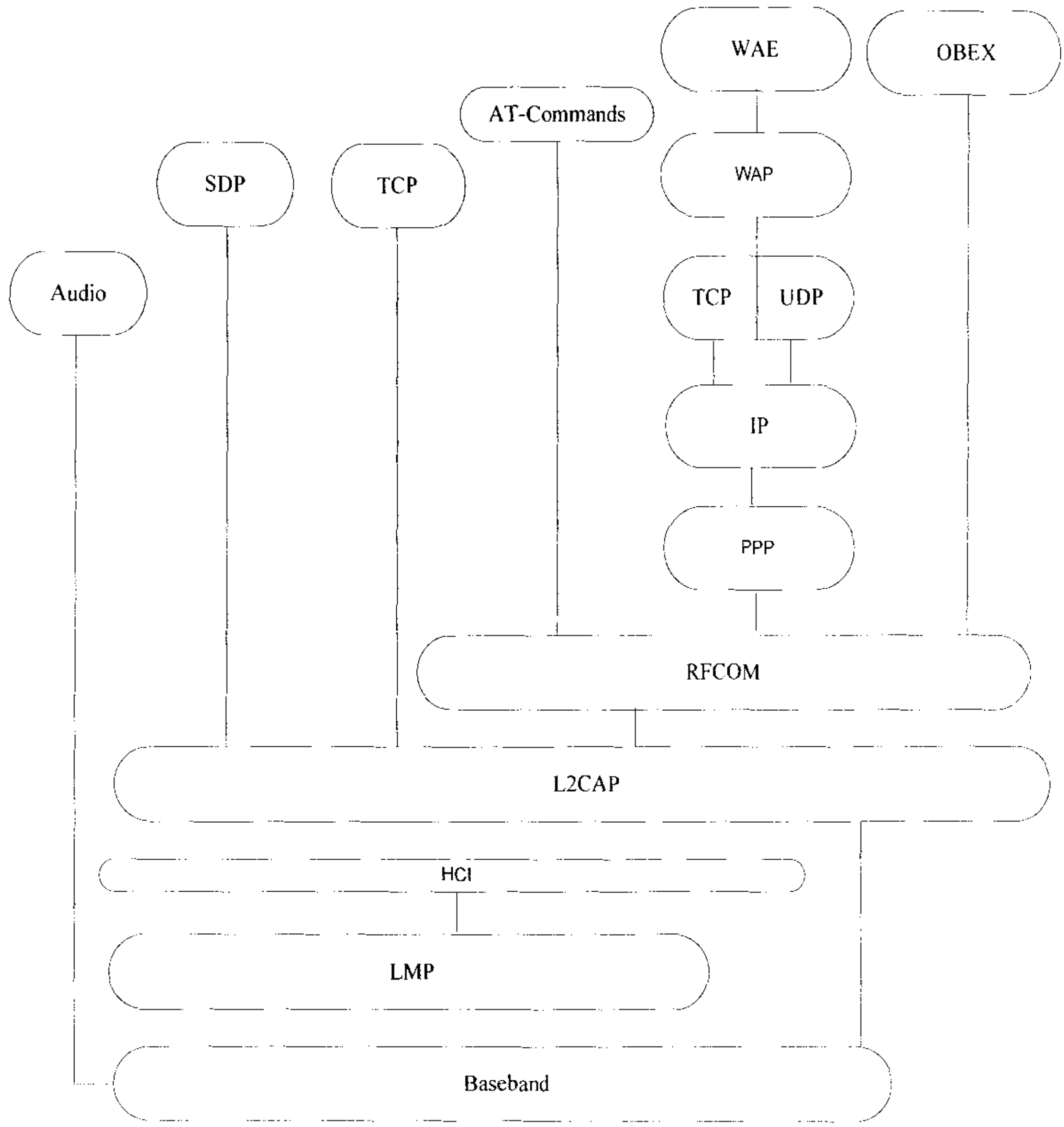


Figure 2.3: The Bluetooth protocol stack

The Bluetooth protocol stack consists of the following four layers (Cordeiro et al. 2003):

2.2.1 Bluetooth Core Protocols

Bluetooth uses many protocols. The Bluetooth core protocols are described by the Bluetooth SIG (Bluetooth_specificaiton 2010; Peters and Heuer 2008). The following are the Bluetooth protocols.

2.2.2 Baseband (BB)

Baseband is the component of the Bluetooth system that identifies the process of medium access and the physical layer to hold up the exchange of the voice, flow of data information and ad-hoc networking among Bluetooth devices (Wei-Yi et al. 2010). The BB and link control layer permit the Bluetooth RF connection between Bluetooth devices to make a piconet. The Bluetooth RF scheme is a Frequency-Hopping-Spread-Spectrum (FHSS) in which data packets are sent in clear time slots on distinct frequencies. Therefore this layer synchronizes the transmission hopping frequency and the clock of the devices by using the inquiry and paging process. It also offers the Synchronous Connection Oriented (SCO) and Asynchronous Connectionless (ACL) packets on the same RF connection. SCO packets contain audio data or a combination of audio and text data, whereas the ACL packets contain text data (Khoutaif and Juanole 2006).

2.2.3 Link Manager Protocol (LMP)

The Link Manager Protocol is used to manage the functions of the Bluetooth connections. This protocol is basically responsible for the link setup, control of logical transports, logical connections, the control of physical connections and security aspects between devices. In addition, it manages the power modes and the connection states of the Bluetooth devices. The LMP is used for the communication between the link managers of two Bluetooth devices that are connected through the ACL logical transport. Each LMP message can be applied exclusively to the physical connection, connected logical links and logical transports between source and destination devices. The LMP messages are not directly transmitted to upper protocol layers (Khoutaif and Juanole 2006).

2.2.4 Logical Link Control and Adaptation Protocol (L2CAP)

The Bluetooth L2CAP adjusts the upper layer protocols on the BB. L2CAP supports only ACL links; it does not support SCO links. It permits upper layer protocols to broadcast and receive its data packets of 64kb, and it passes the connection-oriented and connectionless data services to the upper layer protocols. This protocol sends packets to the Host Controller Interface HCI or to the hostess system in a straight line to the Link Manager. Its link is setup after the ACL link has been established. L2CAP performs some actions:

a) Data multiplexing between upper layer protocols.

b) Segmentation and packet reassembly.

c) Data Multicasting from one way transmission to a cluster of other Bluetooth devices.

d) Quality of Service QoS, provision to the upper layer protocols but no execution of the flow control (Moron et al. 2009).

2.2.5 Service Discovery Protocol (SDP)

The Service Discovery Protocol executes on the top of the Bluetooth stack that handles the publishing and discovery tasks. The more critical services of the Bluetooth structure are discovery services. The SDP presents the foundation for all of the models. By using the SDP, device information services and the attributes of the services can be checked and after that a communication link between the Bluetooth devices can be created (Peters and Heuer 2008).

2.2.6 Cable Replacement Protocol

The Serial Cable Emulation Protocol is used as the cable replacement protocol which is described below.

2.2.6.1 Serial Cable Emulation Protocol (RFCOMM)

RFCOM is a straightforward set of transport protocols and is placed on the top of the L2CAP protocol. This protocol is based on the ETSI standard TS 07.10 and it gives the emulated RS-232 serial ports. The RFCOMM protocol can provide up to 60 simultaneous connections between two Bluetooth devices. The main function of RFCOMM is providing a consistent data stream to the user (Moron et al. 2009; Peters and Heuer 2008).

2.2.7 Telephony Control Protocols

These protocols are used to establish and control the data calls of Bluetooth devices. The following are the telephony control protocols:

2.2.7.1 Telephony Control – Binary

The Telephony Control protocol - Binary (TCS Binary or TCS BIN) is known as a bit oriented protocol. It describes the call control signaling for the organization of speech and data calls among Bluetooth nodes. Furthermore, it describes the process of mobility management for managing Bluetooth TCS. TCS Binary is mentioned in the Bluetooth Telephony Control protocol Specification Binary (Wei-Yi et al. 2010).

2.2.7.2 Telephony Control – AT Commands

The Bluetooth SIG has described the group of the AT commands. A mobile phone and modem can be handled by using these commands. In Bluetooth technology, AT commands are based on the ITU-T Recommendation version V.250 and ETS 300 916 (GSM 07.07). These commands are also used for FAX services (Bluetooth_specificaiton 2010).

2.2.8 Adopted Protocols

Adopted protocols are Point-to-Point, TCP/UDP/IP, Object Exchange Protocol and Wireless Application Protocol which are described below.

2.2.8.1 Point-to-Point Protocol (PPP)

Point-to-Point Protocol executes on the RFCOM in order to achieve the point-to-point connections. PPP is the resource of taking IP packets from the PPP layer and inserting them onto the local area network (Moron et al. 2009).

2.2.8.2 TCP/UDP/IP

TCP/UDP/IP protocols are standard protocols and considered the most widely used protocols which are described by the Internet Engineering Task Force. These protocols are used for communication through the internet. The execution of these protocols in Bluetooth enabled devices permits the communication with other Bluetooth devices linked to the internet (Johansson et al. 2002).

2.2.8.3 Object Exchange Protocol (OBEX)

The Object Exchange protocol is a session protocol built up by the Infrared Data Association (IrDA) and it is executed on top of the Winsock over Bluetooth and IrDA transports. It is used to exchange the objects in an uncomplicated and unstructured way. The OBEX protocol also presents a model for presenting objects and operations.

2.2.8.4 Wireless Application Protocol (WAP)

The Wireless Application Protocol is an open standard to offer mobile consumers access to telephony and information services. WAP works for many wide area wireless network technologies. The main objective of WAP is to transport the internet contents and telephony services to digital cell phones and other wireless devices.

2.2.9 Link controller operation

The link controller operation illustrates the establishment of an intra-piconet and describes how Bluetooth devices can join and leave the piconet. Bluetooth technology has defined multiple states for Bluetooth devices which support the functions of the network formation. This section also discusses the scatternet formation (Moron et al. 2009).

2.3 OVERVIEW OF BLUETOOTH DEVICE STATES

The link controller operation contains three main states: STANDBY, CONNECTION and PARK; it also contains seven sub states: page scan, page, inquiry scan, inquiry, master response, inquiry response and slave response. The sub states are used for the device detection and connection establishment procedure. To switch from one state to another or from one sub state to another sub state, different commands are used. Fig.2.4 shows the different states of link control.

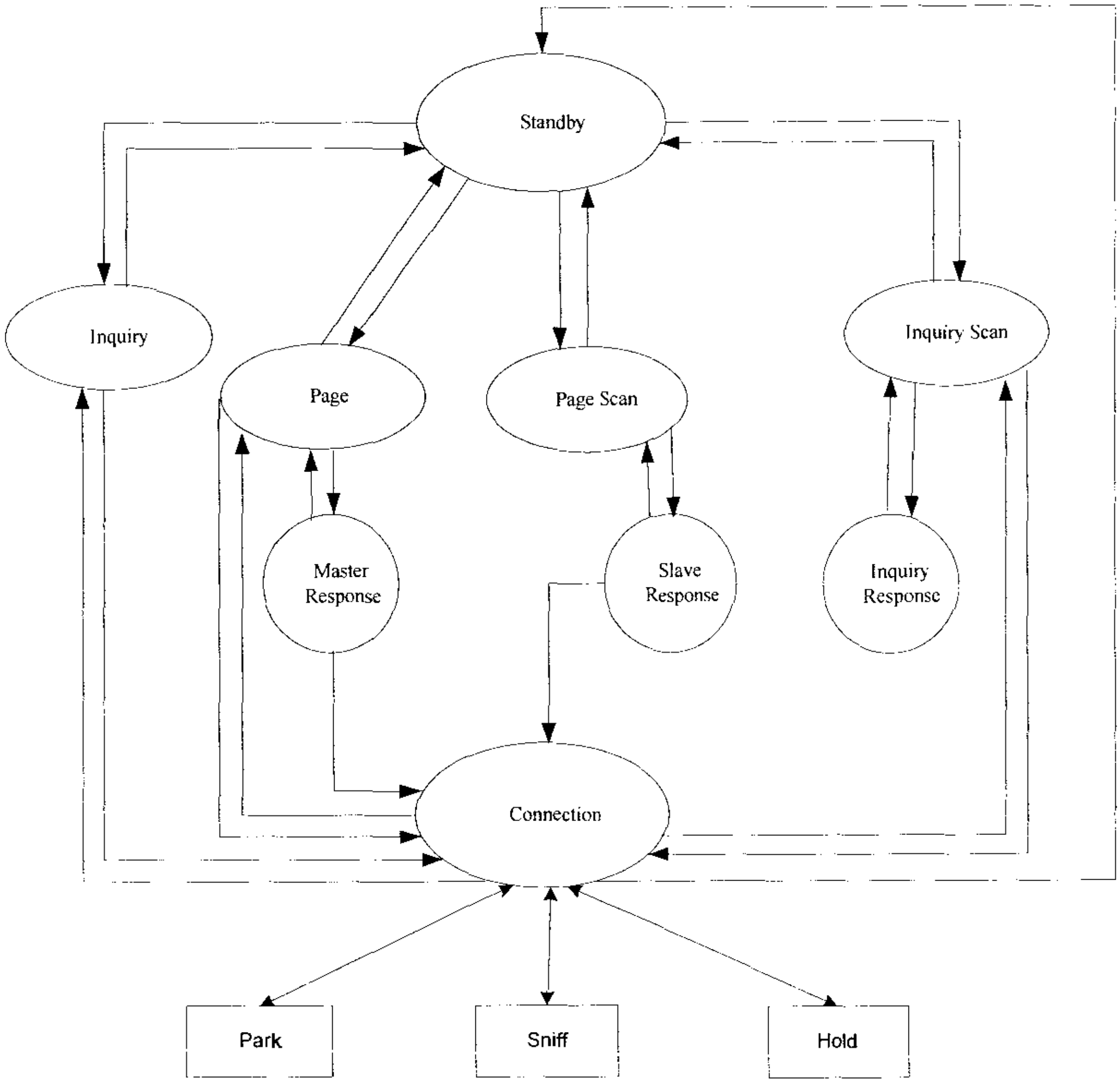


Figure 2.4: Bluetooth device states

2.3.1 Standby state

It is the default and low power consumption state in which only the native clock (CLKN) runs. The Bluetooth clock monitors the timing and hopping sequence of the transceiver. It is typically executed as a 28-bit wrap-around counter. The least significant bit of the clock marks every 312.5μs. The clock has special forms CLKN (native clock), CLKE (estimated clock) and CLK (master clock). There are four phases in a Bluetooth system, which are important in generating certain procedures. These periods are: 312.5μs, 625μs, 1.25ms, and 1.28s as shown in Fig.2.5. These clock periods are represented as CLK0, CLK1, CLK2 and CLK12 (Lo Bello and Mirabella 2006).

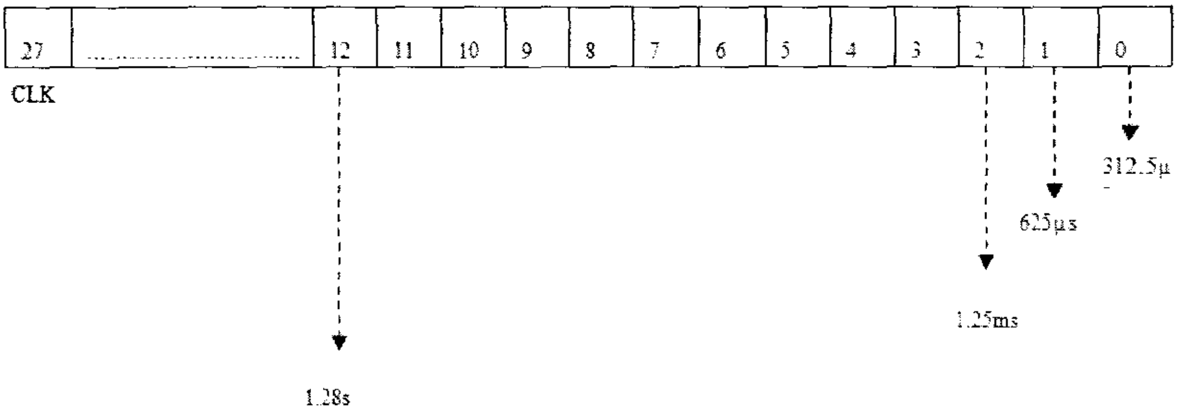


Figure 2.5: Bluetooth clock

Firstly, all Bluetooth devices are considered in the STANDBY mode which means that the devices are not connected with any Bluetooth network (Chakraborty et al. 2010).

2.3.2 Device detection sub states

In order to detect the other Bluetooth devices, a device goes through the inquiry process. During this process, it frequently broadcasts the inquiry messages at different frequencies. A Bluetooth device that wants to be detected frequently performs the inquiry scan operation to respond to the inquiry message. Throughout the inquiry state, the discovering device gathers the Bluetooth node addresses and clocks that respond to the inquiry message. During this process of inquiry and inquiry scan, the native clock of the device is used (Chang et al. 2006).

2.3.2.1 Inquiry scan

The inquiry scan is a process where Bluetooth devices listen for inquiry messages received on their inquiry scan physical channel. This inquiry scan sub state is the same as the page scan sub state. A Bluetooth device in the inquiry scan state decreases its hopping frequency; this state consents to the inquirer to catch with the transmit frequency of the Bluetooth device which is in the inquiry scan state. As the frequencies match, the scanned devices behave like slave devices and transmit their ID and clock information to the master device (Mei et al. 2003). Any Bluetooth device can switch its state from the STANDBY or CONNECTION state to the inquiry scan state. If a device switches its state from the STANDBY state, then it can utilize the full capacity for carrying out the inquiry scan operation. The time interval between the starting of two consecutive page scan operations is considered as the scan interval time $T_{\text{inquiry scan}}$ that should be less than or equivalent to 2.56sec (Cui et al. 2010).

2.3.2.2 Inquiry

The inquiry sub state is the same as the page sub state. This state helps to detect the other Bluetooth devices. In this step, a master device sends the requests to the other Bluetooth devices to make a piconet. The devices which agree to make a piconet, reply to the master's request. The master device sends the ID packets on different frequencies and waits for the reply packets called Frequency Hopping Sequence (FHS) packets from the slave devices. The device sending a FHS packet also sends it's ID and clock value. A Hopping Sequence contains two sets of frequencies called train A and train B (both are 16 frequencies long) (Gelzayd 2002). After getting the FHS packets, the master device carries out the next step called paging. Any node can enter into the inquiry state from STANDBY or CONNECTION state. For the inquiry hopping sequence, the inquiry hopping sequence collection is set as an inquiry. The native clock, *CLKN*, of the inquirer is used. The inquiry hopping sequence can be calculated through equation 2.2 as follows:

$$X_i = [CLKN_{16\text{-}12} + k_{offset} + (CLKN_{4\text{-}2,0} - CLKN_{16\text{-}12})\ mod\ 16]\ mod\ 32 \quad (2.2)$$

where X_i is the inquiry hopping sequence of the X- input and clock-offset as follows:

$$k_{offset} = \begin{cases} 24 & A - train \\ 8 & B - train \end{cases} \quad (2.3)$$

2.3.2.3 *Inquiry Response*

A device in the inquiry response sub state switches from the inquiry scan sub state when it replies to the master inquiry by transmitting its ID and clock status. The slave response for inquiries is completely different from the slave response for paging. During the inquiry scan sub state, when an inquiry message is received, then the receiver returns an inquiry response FHS packet that consists of the receiver's BD_Addr and other parameters. After getting the Inquiry response, a connection is setup for the paging process (Chakrabarti et al. 2004). For X-input, the hopping sequence for the inquiry response is the same as in the slave page response hopping sequence. Equation 2.4 is used for the inquiry response.

$$X_{ir} = [CLKN_{16\quad 12} + N]\ mod\ 32 \tag{2.4}$$

where, the X_{ir} is the inquiry response hopping sequence of the X-input. The N has no start value because it is independent of the consequent value of the inquiring device.

2.3.3 Connection setup sub states

The paging process is used to setup a new connection between Bluetooth devices. For this purpose, the Bluetooth device address is required. The clock information can be obtained from the inquiry process. A device that wants to setup a connection will initially perform the paging process and turn out to be the master device (Jung et al. 2007).

2.3.3.1 *Page Scan*

A Bluetooth device in the inquiry scan state decreases its hopping frequency; this state consents to the inquirer to catch with the transmit frequency of the Bluetooth device which is in the inquiry scan state. As the frequencies matches, the scanned devices behave like slave devices and transmit their ID and clock information to the master device (Chakrabarti et al. 2004; Mei et al. 2003). In this state, a Bluetooth device listens for its Device Access Code (DAC) and responds. The page hopping sequence consists of 32 paging frequencies where after 1.28s a new, different frequency is selected. Two parameters are used to describe the page scan: the scan interval time $T_{page\ scan}$, and the page scan window $T_{page\_scan\_window}$. The time interval between the starting of two consecutive page scan operations is considered as the scan interval time $T_{page\ scan}$. Throughout a scan window, the device listens on one frequency. Any Bluetooth device can switch its state from the STANDBY or CONNECTION state to the page scan state. If a device switches its state from the STANDBY state then it can utilize the full capacity for carrying out the page scan operation (Ching Chu Chang et al. 2007; Kapoor et al. 2004; Mei et al. 2003). During this state, the average energy consumption can be calculated by the following equation 2.5.

$$E_{ps} = T_{page\text{-}scan\text{-}window} (E_{rx\_idle} + (T_{page\text{-}scan} - T_{page\text{-}scan\text{-}window}) E_{sb} / T_{page\text{-}scan}) \quad (2.5)$$

For the scan phases, the energy consumption per time slot is denoted as E_{rx}.. For the duration of the sleep phase, the energy consumption per time slot is denoted as E_{sb} (Leopold et al. 2003).

2.3.3.2 *Page*

If any Bluetooth device needs to make a link with another Bluetooth device, it starts the paging process. The master device performs the paging process; the term paging means the transmission of an ID packet along with a Device Access Code (DAC) again and again until the transmitting device receives a response. The master device does not know on which hop frequency the slave device will wake up so that's why it broadcasts a train of the same DACs on different hop frequencies and listens for a reply. The master device utilizes the BD_Addr and the clock of slave device to obtain the page hopping sequence. Once a master device receives the ID and clock values of the slave devices, then it can easily make a connection with slave devices and communicate (Chakrabarti et al. 2004; Chang et al. 2006). Though, the master and slave devices utilize the same frequency hopping sequence, they never choose the same frequency.

There are 32 page frequencies with 1.28s interval. There are two frequency trains *train A* and *train B* mentioned in equations 2.6 and 2.7.

$$A\text{-}train\ \{f(k-8) \ldots f(k) \ldots f(k+7)\} \tag{2.6}$$

$$B\text{-}train\ \{f(k+8), \ldots, f(k+15), f(k-16), \ \ldots, f(k-9)\} \tag{2.7}$$

If the sequence selection is set to the page state, then the paging device uses the *A-train* $\{f(k-8) \ldots f(k) \ldots f(k+7)$, where $f(k)$ is the receiver frequency of the paged device, the key k indicates the input functions. As in equations 2.8 and 2.9, to get the 8 offset from the *A-train*, a 24 constant is added to the clock bits that is equal to 8 due to the modulo 32 operation. Whereas, the *B-train* is achieved by using the 8 offset.

$$X_p = [CLKE_{16\ \ 12} + k_{offset} + (CLKE_{4\ \ 2,0} - CLKE_{16\ \ 12}) \bmod 16] \bmod 32 \tag{2.8}$$

where

$$k_{offset} = \begin{cases} 24 & A - train \\ 8 & B - train \end{cases} \tag{2.9}$$

2.3.3.3 *Page Response*

When a slave device successfully receives the page message, then the master and slave devices switch to the page response sub state. In order to carry on the connection, they exchange the very important information by using the same channel access code, the same channel hopping sequence and clocks. The channel access code and channel hopping sequence are obtained from the BD_Addr of the master device and the timing can be determined from the clock of the master device. An offset is attached with the native clock of the slave device for the time being synchronised with the slave clock to the clock of the master device (Naik et al. 2005; Whitaker et al. 2005). The slave page response hopping sequence can be calculated by the following equation 2.10 for the X input:

$$X_{prs} = [CLKN^*_{16\ \ 12} + N] \bmod 32 \tag{2.10}$$

The four bits, $CLKN_{16-12}$, will be frozen and these frozen bits are denoted by the asterisk (*). Suppose N is a counter value that starts from zero and it increases by one for each time when $CLKN_1$ is set to zero that matches to the initiate of a master TX slot.

The master page response hopping sequence can be gained by the following equation 2.11 for the X input:

$$Xprm=[CLKE^{*}_{16-12} + k_{offset}^{*} + (CLKE^{*}_{4-2,0} - CLKE^{*}_{16-12}) \bmod 16 + N] \bmod 32 \quad (2.11)$$

The master device freezes its predictable slave clock to the value that triggered a reply from the paged device. It is equal to using the clock value's estimation when receiving the slave response. The frozen clock value is used at the content where the recipient's access code is identified. Let N be a counter that starts from zero and it increases by one for each time when $CLKN_1$ is set to zero that matches to the start of a master Transmitter (TX) slot. Fig. 2.6 shows the general block diagram of the hop selection scheme. The selection box uses clock, frozen clock, N, k_{offset}, address, AFH (Adaptive Frequency Hopping) channel map and sequence selection as input. The address input contains 28 bits and it is selected as the UAP/LAP. In order to select the channel hopping sequence, the AFH channel map is used as an additional input that shows for the RF channels which will be used or unused. The RF channel index output represents a pseudo-random sequence(Cordeiro et al. 2003).

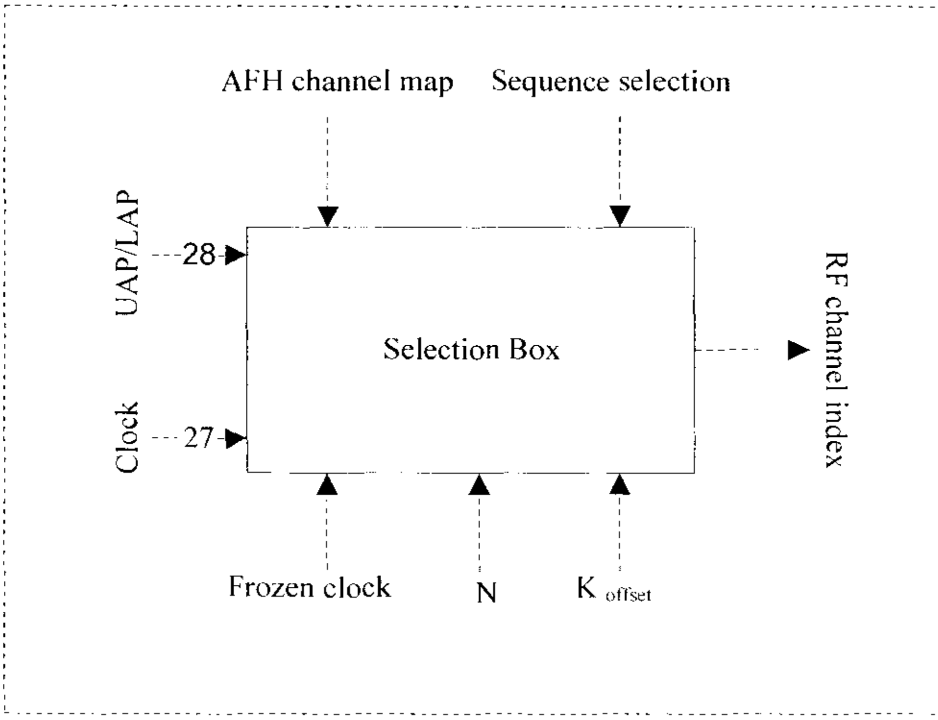


Figure 2.6: Block diagram of hop selection scheme

2.4 CONNECTION STATE

The Bluetooth device in the connection state means that the device is connected with the Bluetooth network and data packets can be transmitted. Any Bluetooth device in standby or connection mode can move to the inquiry state if the device wants to locate another device. In the same way, any Bluetooth device in standby or connection mode can move to the inquiry scan mode if that device wants to be located and connected with another device. In this state the master device transmits a POLL packet to confirm its clock and channel frequency hopping and a slave device can reply by any type of packet. In case the slave device cannot receive the POLL packet and the master device also could not get any reply packet, then both devices will go back to page or page scan sub states (Chakraborty et al. 2010; Kapoor et al. 2004; Vojislav et al. 2003). The connection states and modes are shown in the following Fig.2.7.

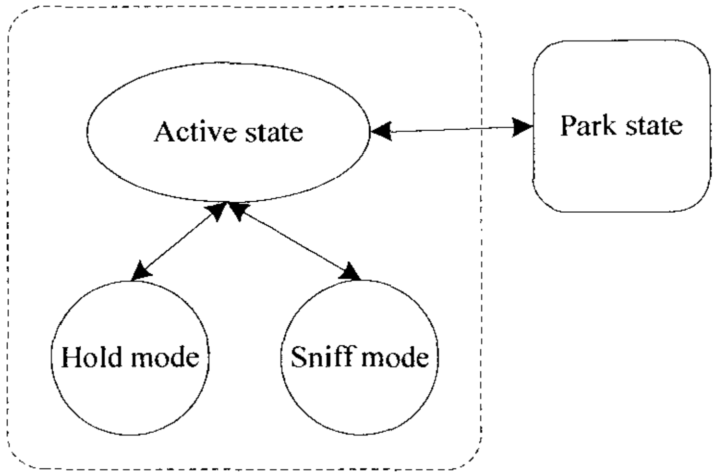


Figure 2.7: Connection states and modes

2.4.1 Active state

During the active state, Bluetooth devices can actively contribute on the channel. In a piconet, a maximum of seven slave devices can be active in a given time. The master device schedules all the data traffic to and from the slave devices. Active slave devices listen for the packets from the master device and if the slave devices are not addressed then they sleep until the next transmission is received (Alkhrabash and Elshebani 2009).

2.4.1.1 Park state

A slave device goes into the park state if it does not want to contribute in the traffic transmission but wants to be synchronized. In this state, the slave device transmits its AM_Addr but receives a Parked Member Address (PM_Addr 8 bits) and Access Request Address (AR_Addr 8bits). A parked slave device wakes up at all intervals and listens to the channel to be re-synchronized and to communicate (Yu et al. 2009).

2.4.1.2 Sniff mode

Bluetooth devices can save their battery power by using the sniff mode. To go into the sniff mode, the master device sends a sniff command through the Link Manager (LM) protocol that includes the sniff interval T_{sniff} and an offset D_{sniff}. The sniff slots reserved for the master to slave will start if the clock satisfies one of the following equations, 2.12 or 2.13.

$$CLK_{27\text{-}1} \; mod \; T_{sniff} = D_{sniff} \tag{2.12}$$

Or

$$(CLK_{27}, CLK_{26\text{-}1}) \; mod \; T_{sniff} = D_{sniff} \tag{2.13}$$

Sniff mode basically reduces the duty cycle of the slave device listening activity. It is the least power efficient mode as compared to the other power saving modes (hold and park) (Blom et al. 2008). The sniff mode negotiation process can be started by a slave or master device. Fig.2.8 shows the sniff mode operation.

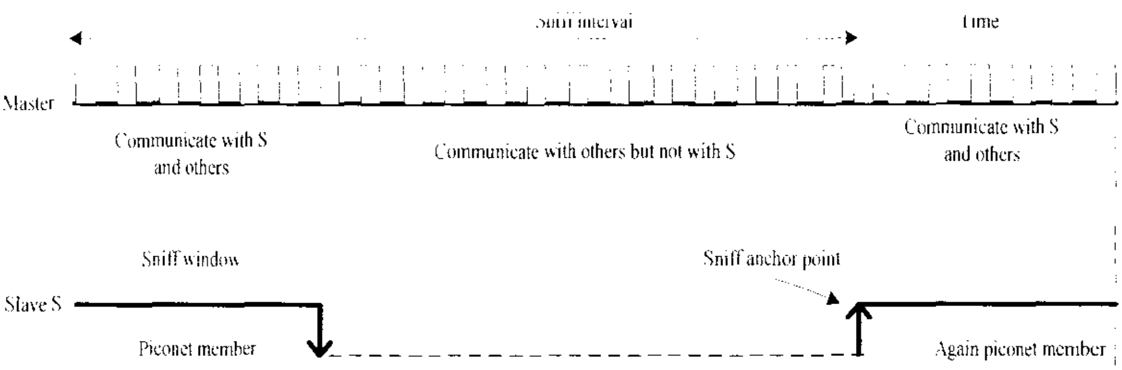


Figure 2.8: The sniff mode operation

2.4.1.3 Hold mode

The master device has the ability to set the slave devices into the hold mode where only the internal clock runs. A slave device can also demand to be set into the hold mode. It is a medium power efficient mode because of having an intermediate duty cycle (Yu et al. 2009). For the duration of the hold mode, the ACL link to a slave device is temporarily not supported. A device in hold mode keeps its AM_ADDR and resumes the data transfer when the device is out of the hold mode (Ching-Fang and Shu-Ming 2008). In the hold mode, the master device does not transmit the POLL messages to the slave device for a specific time period known as a hold time out. The time period of the hold mode is consulted between the master and the slave devices. Any device (master/slave) can start this process. The device which initiates the process, suggests switching the hold mode and hold time out. The other device acknowledges it (Kapoor et al. 2004). Fig.2.9 shows, how the messages are switched over between the devices.

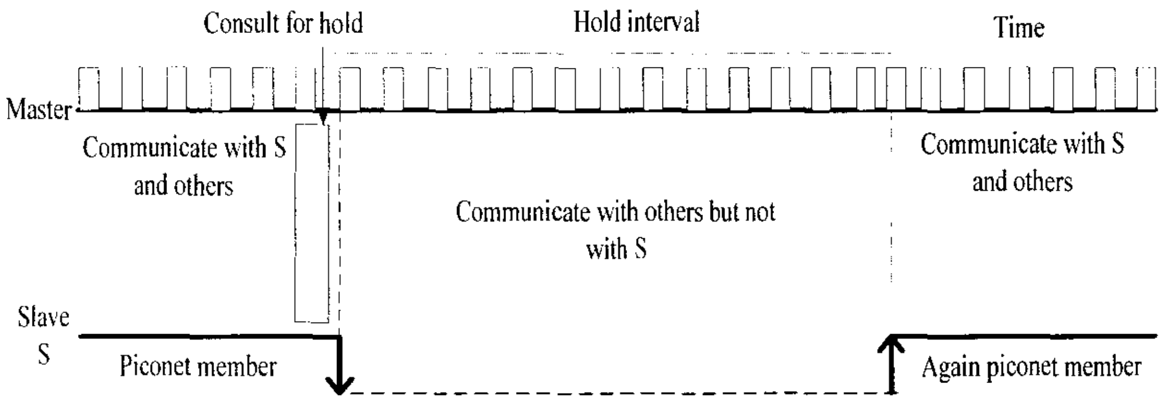


Figure 2.9: The operation of hold mode

2.5 BLUETOOTH PACKET STRUCTURE

In order to exchange the information between devices, data packets are used on different hop frequencies. Fig.2.10 shows the Bluetooth packet structure that consists of the following three parts (Naik et al. 2005):

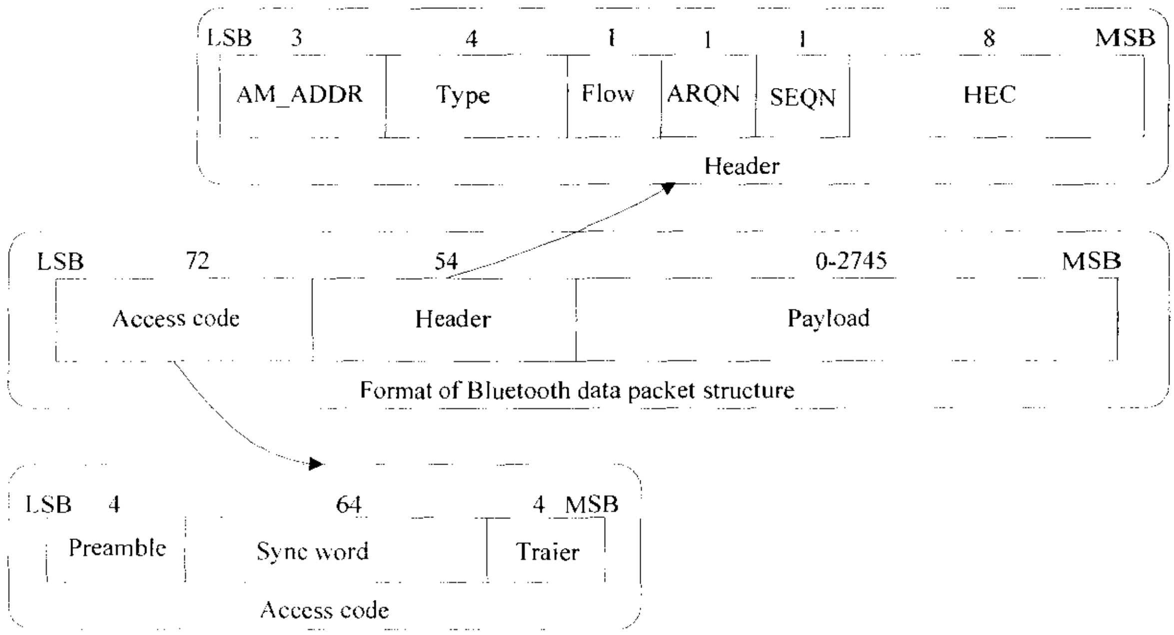


Figure 2.10: Bluetooth data packet structure

2.5.1 Access Code

All access codes are obtained from the LAP of the BD_Addr. The access code is used in piconet formation protocols as classification of the devices which are engaged in the piconet formation process. An access code of all of the packets will be identical in a piconet. If the access code is followed by a header, it consists of 72 bits or else it will consist of 68 bits. The access code supports bit and word synchronization. Moreover, the access code identifies the receiver device about the arrival of the data packets. As well, it is used for offset compensation and time synchronization. An access code consists of the following types:

a) Device Access Code (DAC)

The device access code is used for the particular signaling processes, e.g., paging and response to the paging, and it is obtained from the BD_Addr of the paged device and is used for the duration of page scan, paging and page response sub states.

b) General Inquiry Access Code (GIAC)

The GIAC is general to everyone's device and it is used to find out the other Bluetooth devices which are in the proximity of other devices.

c) Dedicated Inquiry Access Code (DIAC)

The DIAC is general for a dedicated cluster of Bluetooth devices that shares a common feature. The DIAC can be used to find out the dedicated Bluetooth devices which are in the proximity(Shih et al. 2006).

d) Channel Access Code (CAC)

The CAC is derived from the master's BD_Addr and it contains the preamble, sync word and trailer. The CAC describes the piconet channel and structures the preamble of all of the packets switched on the channel. Its length is 72 bits (Misic et al. 2006).

2.5.2 Header

The header is also a part of a packet structure that describes the target device. When devices communicate then the header acknowledges the packet information between a pair of devices. The length of the header is 54 as shown in Fig.2.10. A header consists of the following six parts:

a) Active Member Address (AM_ADDR)

The AM_ADDR stands for a member address. It is used to differentiate the active members contributing in the piconet (Naik et al. 2005). As in a piconet, one or more than one slave devices can join to a master device so to recognize each active slave device individually, each slave device is allocated a 3-bit temporary address. All the packets exchanged between the master-slave or slave-master communication carry the AM_ADDR of the slave devices.

b) Type

The 4-bit Type field describes the form of that packet which is transmitted. The Type field defines the 16 different packet kinds. It specifies the packet slot figure (how many slots the current packet will reserve) and data type (Synchronous SCO or Asynchronous ACL) inside the payload field (Bluetooth_specificaiton 2010).

c) Flow

Flow is used to control the packet flow over the ACL link. Flow zero indicates that the receiving buffer is full or prevents the transmission of the data, temporarily. The stop signal only related to the ACL data packets. The SCO packets or packets containing the link control information like ID, Null and POLL can be received. A GO indication (Flow=1) is returned if the receiving buffer is empty (Moron et al. 2009; Tekkalmaz et al. 2006).

d) ARQN

The Automatic Repeat reQuest Number (ARQN) is a 1-bit acknowledge indication that reveals the achievement of the packet transformation. The acknowledge can be positive acknowledge (ACK) or negative acknowledge (NAK). An ACK (ARQN=1) will return if the positive response is received or else NAK (ARQN=0) will return. A NAK acknowledge is considered if no acknowledgement is received (Kleinschmidt et al. 2009; Ramana Reddy et al. 2010).

e) SEQN

In order to stream the data packet, the SEQN bit gives a sequential numbering system. This is necessary to filter out the retransmissions of data to the destination. If the retransmission of the data packets happens because of a failure of the ACK, then the target device gets the same packet twice. By the comparison of the SEQN of repeated packets, retransmission of packets can be discarded.

f) HEC

The Header-Error-Check (HEC) consists of an 8 bit word produced by the polynomial 647 (octal representation). The HEC ensures the header integrity. The slave Upper Address Part (UAP) is used for the transmission of FHS packets in a master page response state. The Default Check Initialization (DCI) is used for the transmission of FHS packets in the inquiry response state. The UAP of the master device is used for all other cases. After initialization, the HEC is considered for the 10 header bits. Before the examination of the HEC, the receiver has to initialize the HEC check with the correct 8-bit UAP or DCI. The entire packet will be ignored if the HEC packet cannot be checked (Khan et al. 2009).

2.5.3 Payload

The payload fraction of a packet consists of two subparts: the data and the payload header. The payload header may consist of two sizes depending on the packet size (time slots that it reserves). The payload header will be one byte long for single slot packets; for the multi slot packets, the payload header length will be two bytes long. The payload header keeps the information of the data length (Ramamurthy et al. 2007).

2.6 TAXONOMY OF BLUETOOTH NETWORK TOPOLOGIES

The Bluetooth standard permits the formation of the piconet, which contains one master device with seven active slave devices. Any additional slave devices must be in parked mode. They may create an overhead because of parking and un-parking them. Even though the Bluetooth standard permits for the multiple connected piconets known as a scatternet, it does not provide any specific scatternet protocol. The following is the classification of the scatternets (Stojmenovic and Zaguia 2001).

2.6.1 Single hop Bluetooth scatternets

In a single-hop scatternet, all Bluetooth scatternet devices are considered in a radio range of each other. It contains two important types: a centralized Bluetooth scatternet and a distributed Bluetooth scatternet which have further sub types.

2.6.1.1 Centralized Bluetooth scatternets

The centralized Bluetooth scatternet protocols are categorized into the following protocols.

a) Supper master selection based scatternets

In this type of scatternet, firstly, the Bluetooth scatternet construction algorithm uses an inquiry process to gather the neighborhood information. The sender device looks for receivers on randomly selected frequencies and the identified receivers reply after a random back-off time delay. During this process, a supper master is selected from the masters that gathers all network information and assigns the tasks to the connected devices. Although, this technique is a centralized technique, it does not provide scalability (Salonidis et al. 2001).

b) Ring topology

Ring shaped scatternets are defined for a single hop scatternet (Lin et al. 2003) where all of the master devices generate a ring. There may be some slave devices that can connect with the master devices outside of the ring. The ring shaped scatternet is a very simple network for routing that reduces the bottleneck problem. Ring topology can be a master/ slave ring or a slave/slave ring. Fig.2.11(a) shows the master/slave ring and Fig.2.11(b) shows the slave/slave ring. The problems with this topology are: it is centralized but not scalable and it uses the park mode that creates the delay (Persson et al. 2005).

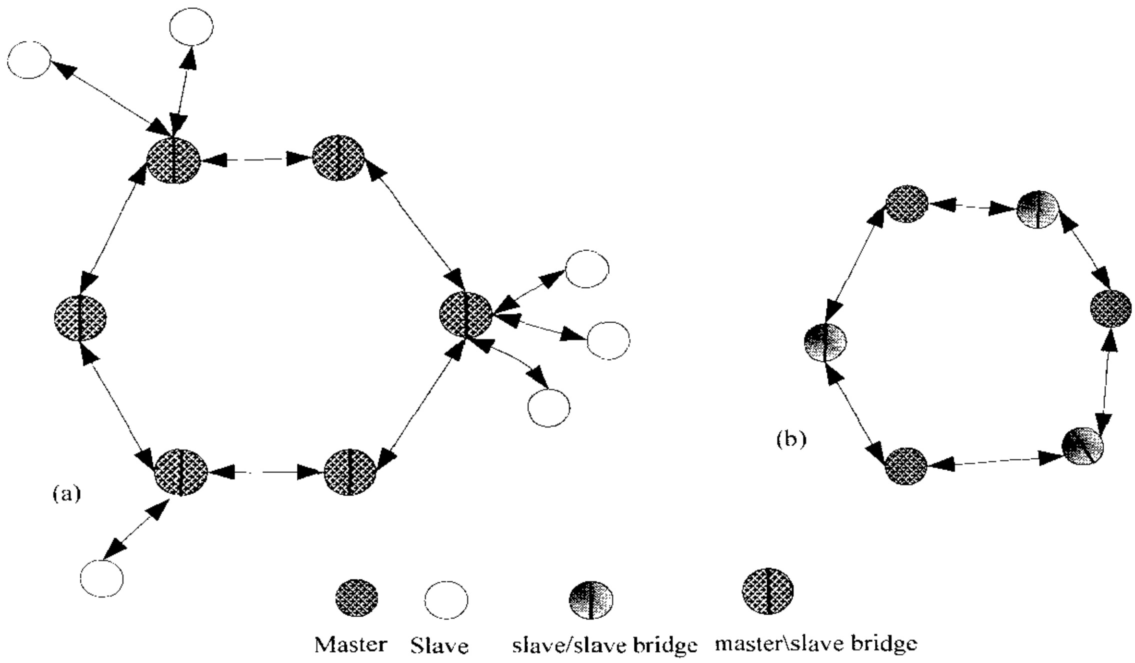


Figure 2.11(a): master/slave ring scatternet formation (b): slave/slave ring scatternet formation

2.6.2 Distributed Bluetooth scatternets

A distributed Bluetooth scatternet contains the following scatternet formats:

a) Tree formation scatternets

It is a tree shaped structure which illustrates the self routing scatternet for a single hop network. Bluetooth devices used in this network are arranged in a search tree format. The Bluetooth ID is used as a key. The tree formation scatternet presents a parent child structure between the devices as in Fig. 2.12. This structure contains one Blueroot device (randomly selected) known as the parent device that constructs a routed spanning tree. The Blueroot is considered as a master device and all of the one hop neighbor devices are considered as slave devices or children. Each internal device performs the function of the master/slave bridge. This scatternet type is useful for short routing links. The shortcoming with this technique is that if the parent device fails it disconnects the other devices (Lin and Wang 2010).

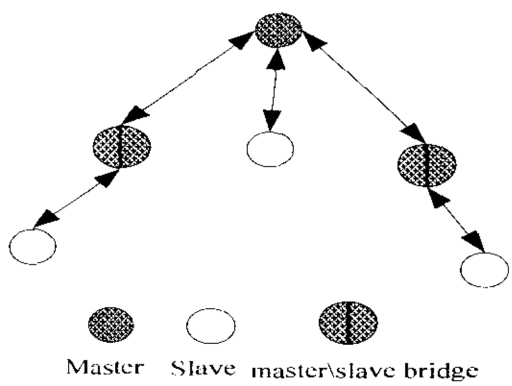


Figure 2.12: Tree formation scatternet

b) Loop formation scatternets

The formation of the loop scatternet is proposed for the smaller networks with a small network diameter. This scatternet formation generates a loop in which the slave/slave bridges are used. This topology is used to minimize the number of piconets and it provides the protection to the connected devices (Stojmenovic and Zaguia 2001).

2.6.3 Multi hop Bluetooth scatternet

The multi hop Bluetooth scatternet contains the following:

2.6.3.1 Centralized Bluetooth scatternet

For multi hop scatternets, a centralized approach is proposed that improves the whole network connectivity and overcomes the problem of congestion on the devices. This approach is helpful for two main reasons:

- Locating the most excellent performance for a given visibility graph.
- Gaining information on scatternet patterns and performance.

A centralized approach which is based on the genetic algorithm chooses an arbitrary set of devices for the initial population. Each set contains the combination of master, slave or bridge devices. The main purpose of this approach is to reduce the number of piconets (Sreenivas and Ali 2004).

2.6.3.2 Distributed tree formation scatternet

A Bluetree scatternet formation for multi hop scatternets is proposed in which each device is assigned limited functions. The protocol of this topology selects a single device known as the Blueroot. This Blueroot performs the function of a master device where each one hop neighbor device becomes its slave device. When new devices join the piconet, then the slave devices (children) will perform the additional function of master. The intermediate device between new slaves and the Blueroot will become the master/slave bridge. This protocol repeats this procedure until all of the devices are assigned their functions (Basagni et al. 2004).

2.6.3.3 Clustering based scatternet formation

In this scheme, a master device is selected as a cluster head. In order to provide the neighbor's information to other devices, this protocol completes the scatternet detection process before the network clustering (Petrioli et al. 2003). In this technique when the device discovery stage completes, every Bluetooth device calculates its weight on the basis of its degree and other parameters and compares them with its neighbors for determining its role (master/ slave). A device that has a higher weight compared to the neighbors will become the master and the remaining devices will become slaves. The master device selects a bridge device to connect with the neighboring piconets. In the resultant, each piconet can contain more than seven slave devices; this can degrade the performance.

2.7 ROLE SWITCHING TECHNIQUES OF BLUETOOTH NETWORKS

The performance of a scatternet is extremely dependent on the efficient communication between Bluetooth devices. For efficient communication, many researchers have used role switching operations. The role switching technique has different types of role switching operations which can be used for different requirements (Klajbor et al. 2010).

2.7.1 Merging role switching operation

In this operation, the devices of two piconets (P_1, P_2) can combine into one piconet. This type of operation reduces the number of bridges and piconets. As shown in Fig.9 (a), two piconets, P_1 and P_2, are connected through an intermediate node (b). Nodes (a, c, f, d, and e) are in the range of node b; according to the role switch operation, node b can perform the function of master and merge two piconets into one piconet. In Fig. 2.13(b), the bridge node becomes the master and masters (c, e) change their roles from master to slave.

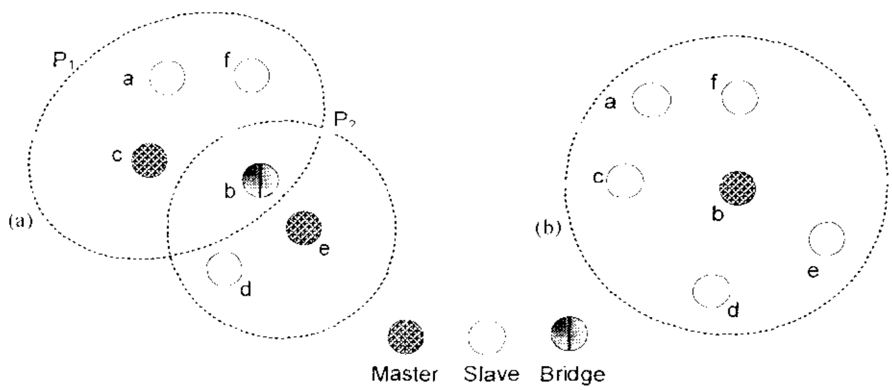


Figure 2.13(a): Before merging the role switching operation (b): After merging the role switching operation

2.7.2 Splitting role switching operation

A role switching operation can divide one piconet into more piconets related to a specific requirement. This operation increases the number of piconets and bridges. For example in Fig.2.14 (a), before executing the splitting role switch operation, there is one piconet having one master with 5 slave nodes. Fig.2.14(b) shows the role switch operation that splits one piconet into two piconets P(1, 2) by changing the role of devices.

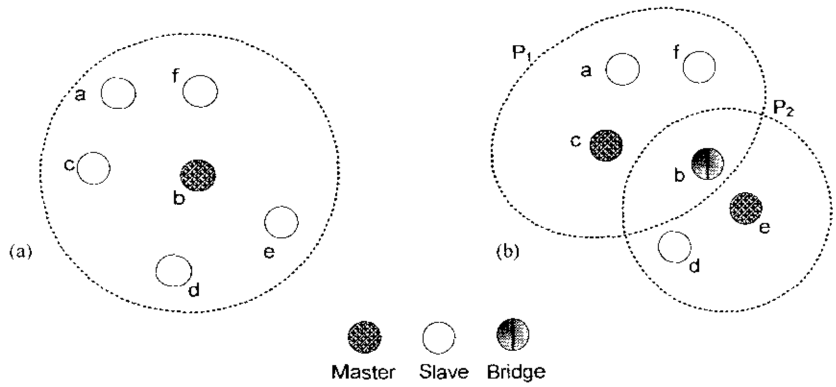


Figure 2.14(a): Before performing the splitting role switch operation (b): After performing the splitting role switch operation

2.7.3 Taking-over role switching operation

The role switch operation can be applied for any Bluetooth device to take-over the resource of another device. During this operation, devices can change their roles from slave to master, master to slave, bridge to master/slave and master/slave to bridge. For example, Fig. 2.15(a) shows that there is one piconet where b is master of five slave nodes. Fig.2.15(b) shows how different nodes can take over the roles of other nodes (Chang et al. 2008b).

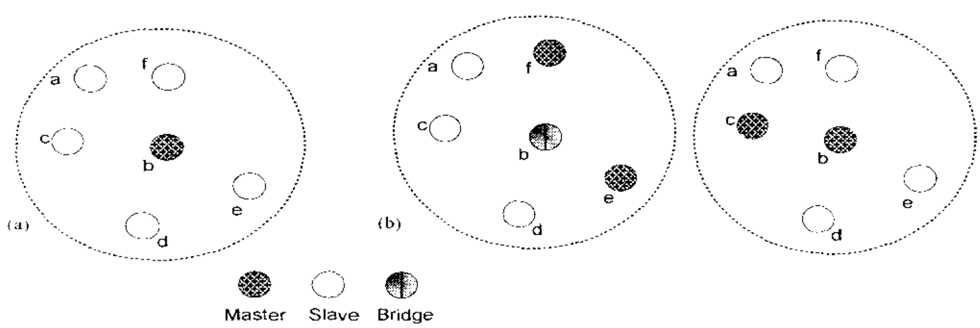


Figure 2.15(a): Before the Take-over role switch operation (b): After the Take-over role switch operation

2.8 BLUETOOTH PROTOCOL PROPERTIES

The properties of the Bluetooth protocol describe its capability of generating and maintaining Bluetooth networks. The following are the main scatternet performance properties:

2.8.1 Time complexity

Time complexity is the total time required for the creation of a connected scatternet. The protocol time complexity regards how much time it will take to create a fully connected scatternet. In order to reduce the delay time, the scatternet should be created quickly (Vergetis et al. 2005).

2.8.2 Message complexity

The message complexity is the total number of messages exchanged between the Bluetooth devices. As Bluetooth devices have limited battery power, it is very important that the maximum number of messages should be transmitted in the less amount of time (Whitaker et al. 2004).

2.8.3 Environmental dynamism

A dynamic Bluetooth network is similar to the ad hoc networks. In order to perform the ad hoc operations in the Bluetooth network, efficient scatternet configuration and maintenance is required. In order to assign the device roles, the protocol should have the ability to make a decision dynamically and in a distributed manner. When Bluetooth devices enter and leave the scatternet at random times the protocol maintains it (Yip and Kwok 2007).

2.8.4 Bluetooth scatternet properties

The scatternet performance can be analyzed by using traffic dependent or traffic independent procedures. The traffic dependent procedures define the source and destination, sequence of data packets, routing techniques and scheduling techniques etc. Sometimes, it creates assessment problems. Therefore, traffic independent procedures are considered (Whitaker et al. 2005).

2.8.5 Lifetime of the Bluetooth network

Bluetooth devices have limited battery power so the Bluetooth devices should be energy efficient because this constraint increases the life time of the Bluetooth devices. In order to handle the data traffic sequence across the scatternet, two power saving procedures\techniques are defined. One process uses the battery power based on the master slave communication and the other technique is based on the distance. An efficient scatternet formation can make an energy efficient scatternet (Tekkalmaz et al. 2006).

2.8.6 Network reliability

The connected scatternet should be reliable. If the protocol which is used to connect the scatternet reacts quickly against any failure of a device or link, then this protocol ensures the reliability of the scatternet. Flat scatternet formation protocols are considered to increase the network reliability. Network reliability plays an important role for efficient communication within the scatternet (Whitaker et al. 2005).

2.8.7 Network diameter

The longest path contains the maximum number of hops between a source and destination that ultimately creates the problem of delay and extra resource utilization. So, the protocol should follow the shortest and most efficient path (Xiang et al. 2009).

2.8.8 Less number of piconets

The number of slave devices states the number of piconets needed within the network. With fewer piconets, the communication within the scatternet can be handled easily. Since each piconet uses its own frequency channel, if more piconets communicate at the same time, then collisions can occur; this degrades the performance of the scatternet (Jingli et al. 2009).

2.8.9 Number of roles per device

After the device initialization, each device is assigned its role (master, slave or bridge). In some cases, due to specific requirements, a device can be assigned multiple roles. If a single device performs multiple roles, then it can degrade the performance of the network because of switching, losing energy etc (Whitaker et al. 2005).

2.8.10 Shortest path length

The routing protocols should follow the shortest path for the communication of a source and destination. The longest path contains some extra devices which consume extra resource utilization that ultimately degrades the performance of the scatternet. An efficient routing technique should be adapted; if multiple routing paths pass through the same devices then they can create a bottleneck. In order to pass the light data traffic, the shortest path measurement is considered as a realistic measure (Xiang et al. 2009; Yip and Kwok 2007).

2.9 BLUETOOTH APPLICATIONS

Bluetooth technology is used in many applications which are categorized into several groups, like workplace applications, ad-hoc networking, access control applications, low bandwidth applications and real-time location systems.

2.9.1 Workplace applications

These applications are associated to the setting up of a wireless workplace environment. For example, by using a Bluetooth link, the mouse and a keyboard are connected with the computer, a Bluetooth enabled printer can connect many computers at a time and a Bluetooth technology can be used for the data synchronization between the Bluetooth enabled devices (Rotaru et al. 2008).

2.9.2 Ad-hoc networking

Bluetooth enabled devices can make a large ad-hoc network commonly named the scatternet (Dunne et al. 2005). The devices in ad-hoc networks can connect and exist as long as required if they are within the radio range of each other. Currently, these applications are used to exchange the electronic business cards between users.

2.9.3 Access control applications

Bluetooth technology can be used in an access card for the purpose of having the right of entry in a building or accessing resources without using a swipe machine. This technology can be used for mobile payment (credit card) (Mei et al. 2003).

2.9.4 Low bandwidth applications

It can be used for the low bandwidth applications wherever high bandwidth is not needed and wireless connections are required (Chang et al. 2008a).

2.9.5 Real-time location system

The real-time location system is used to track the items and it can be used for the identification of the position of the objects in real time with the help of tags or devices. Theses tags or devices are fixed in the objects that receive the wireless signals(Ka-Wai and Luong 2003).

2.10 BLUETOOTH VS. OTHER TECHNOLOGIES

In this section, Bluetooth is compared with the other wireless technologies like ZigBee and Wi-Fi. These are different standards used for wireless communication. They have different hardware requirements. The basic distinctions of these technologies make it impossible to use these technologies for the same functions because they have their own advantages and uses.

Bluetooth technology is more helpful when two or more devices which are in the proximity of each other want to exchange data, like phones, modes, printers and headsets etc. Bluetooth technology is enabled in lots of devices and it has the ability to transfer any kind of data (audio, video and text). It is very easy to setup the Bluetooth devices for communication. After the initialization of the devices they can search and recognize each other. As Bluetooth devices can directly connect and communicate with each other, they can be used anywhere.

Wi-Fi is the abbreviated form of "Wireless Fidelity". It has the ability to transmit data fast because of using wave forms. It can be implemented into the computers and notebooks. A motionless access point is used to connect the Wi-Fi to the internet. An "access point" is additional equipment that the Wi-Fi usually needs to permit devices to connect and communicate. Mostly, the Wi-Fi works in fixed locations known as "hot spots". The speed of the Wi-Fi connection decreases if it connects to the multiple devices by using the same "access point" (Klajbor and Woźniak 2008).

ZigBee is a low-cost and low power wireless network standard. Its low cost permits it to be used worldwide in wireless control and observing applications. ZigBees can sleep most of the time so they consume less energy. The low power consumption increases its battery lifetime. The ZigBee devices activate in 15sec because they enter from sleep to active mode.

Table 2.1: Comparison of the wireless standards

Characteristics	Bluetooth	Wi-Fi	ZigBee
Standard	IEEE 802.15	IEEE 802.11	IEEE 802.15.4
Frequency	2.4GHz	2.4GHz and 5 GHz	2.4GHz
Cost	Low	High	Low
Range	10m	50m-100m	10m-100m
Power consumption	Low	High	Very low
Latency	200ms	150ms	
Ease of use	Simple, make connection between 8 devices at a time	Complex, it needs hardware and software configuration	Coordinator selection
Network topology	Ah-hoc (small networks)	Point to hub	Ad-hoc, star, mesh, or peer-peer
Complexity	High	High	Low
Network size	8	32	2 - 65,000
Application focus	Cable replacement	Web, E-mail & video	Control & monitoring
Primary devices	Mobile phones, mouse, keyboards, office and industrial computer devices etc	Notebooks, desktop computers, servers, TV, and Latest mobiles	Input devices(mice, keyboard etc), Remote Control-equipped HDTVs

2.11 EXISTING RELATED WORK

Many research works have been carried out on Bluetooth inter-piconet communication (Jarrah O.M and Megdadi. 2009; Xiang. L, *et al, 2009;* Yu. C, *et al 2009;* Shafie. A et al. 2008). The inter-piconet communication is still an open research issue as it is not mentioned in the Bluetooth specification. The Bluetooth node connection formation time is too long; therefore, dynamic source routing is preferred for scatternet communication. Conventional ad hoc routing protocols can be used for Bluetooth networks. BlueStar (C et al. 2003) is a scatternet development technique, where each node, after the discovery stage, calculates its weight on the basis of its degree and other parameters and compares them with its neighbors for determining its role as (master/ slave).

The BlueStar protocol sets up its own separate piconets; a node that has a higher weight compared to its neighbors will become the master and the remaining nodes will become slaves. The master node selects a bridge node to connect with neighboring piconets. In the resultant BlueStar scatternet, each piconet can contain more than seven slave nodes. However, it is observed that having more than seven slave nodes in a piconet could degrade the performance. Another scatternet protocol called BlueMesh (C.F. Chisasserini et al. 2003) has been proposed for the improvement of the BlueStar protocol. BlueMesh activates in two stages; in the first stage, it determines one and two hop neighboring nodes and in the second stage, the Bluetooth nodes select their roles. BlueMesh defines a procedure for handling the topology variations based on the insertion and removal of Bluetooth nodes. When a new node wants to join the scatternet, it first broadcasts identity packets and then the receiving nodes reply with a FHS packet. New nodes can receive FHS packets from more than one neighbor; in this case, a decision is required for connection establishment. Few researchers also proposed some techniques for Congestion handling and route maintenance as discussed in detail as below.

2.11.1 Dynamic Piconet Restructuring Protocol

The Dynamic Piconet Restructuring Protocol (PRP) was proposed to share the congestion on the master device. PRP shares the master device congestion by forming new piconets of slave devices that can communicate directly, where one slave device acts as master and others act as slave devices. During the restructuring operation, the slave device with light traffic flow will be selected as a new master device. For example, as shown in Fig.2.16 (a), devices (a, d), (e, f) and (c, b) are communicating through a master device. According to PRP, when a master device detects congestion, it performs piconet restructuring by using the role switching operation. Fig.2.16 (b) shows the piconet structure after the role switching operation.

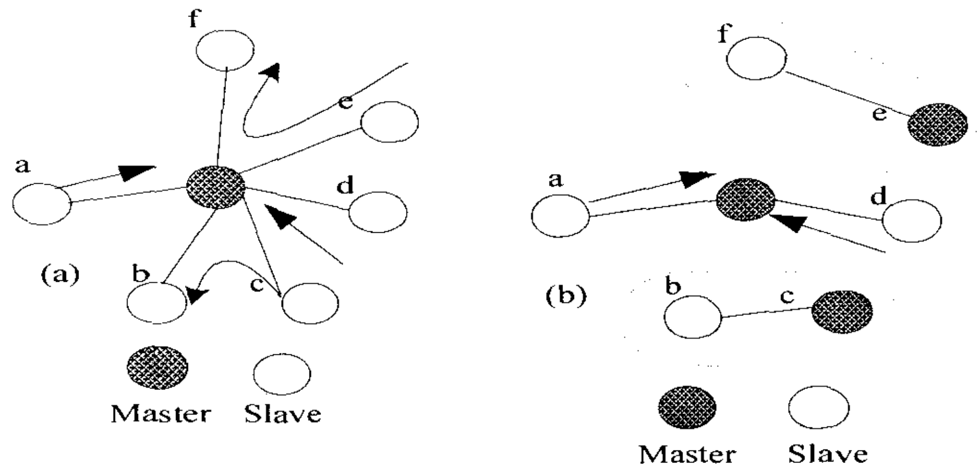


Figure 2.16(a): Traffic flow analysis before restructuring (b): Piconet structure after role switching

Although PRP solves the congestion problem on the master device through traffic load handling, it also creates some serious problems: 1) Loss of active member addresses because of link breakage as PRP makes direct communication links between a source and destination in new piconets and breaks the original links of the devices with the original master. At the same time, if new Bluetooth devices arrive and join the piconet, the master device assigns the active member addresses to the new devices and reaches the limit. After t+1 time, when the devices come back in their original states, they cannot join the piconet. 2) Increasing guard time, as PRP makes unnecessary new piconets within piconets, it wastes synchronization time. PRP makes new piconets for all communicating pairs of source and destination without considering whether they are frequently communicating or not. More unnecessary piconets utilize the extra resources. 3) Increases delay when new devices arrive and join the piconet. If new devices want to communicate with the devices which have changed their roles during restructuring, they will have to wait for the devices to return in their original states.

2.11.2 Dynamic relay reduction and disjoint route construction protocol

The authors in the dynamic relay reduction and disjoint route construction protocol (Yu et al. 2007) optimize the number of bridges. In this protocol, each bridge device starts its transmission by transmitting a request packet known as a Relay Information Collection (RIC) to the master devices it connects for accumulating the information of all relay/bridge devices belonging to its piconets. A relay Connection Table (CT) is created after receiving the reply packets from the master devices. Once the CT is created, then the bridge devices choose the role of the bridge device; they decide if the role of bridge device should change from bridge to slave or not. In Fig. 2.17(a), master M1 transmits a RIC packet to relays $R(P_1,P_2)$, $R(P_1,P_5)$, $R(P_1,P_3,P_5)$, $R(P_1,P_3)$, $R(P_1,P_3,P_4)$ and $R(P_1,P_4)$ and after receiving the RIC packet, each relay transmits a response packet to M_1. As Relay $R(P_1, P_3)$ is connected to the master M_1 and M_3, it confirms the degree of the other relay devices and checks if any other relay device has a connection with M_1 and M_3. If one exists, then the role of that relay device will be changed from relay to slave. Fig.2.17 (b) shows the scatternet after executing the relay reduction protocol; according to this protocol the roles of $R(P_1,P_3)$, $R(P_1,P_5)$, $R(P_1,P_4)$ and $R(P_3,P_5)$ have been changed from relays to slaves.

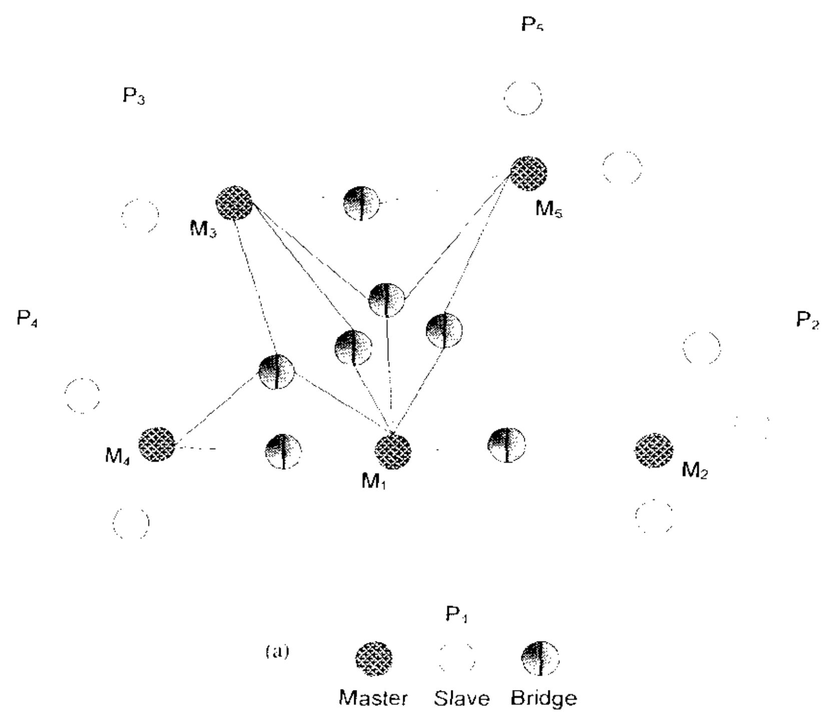


Figure 2.17(a): A Scatternet before executing the relay reduction

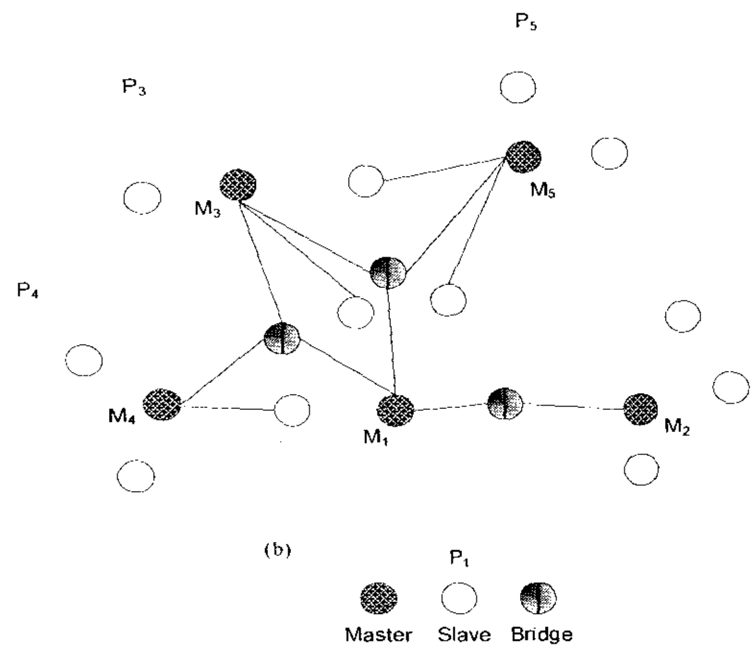


Figure 2.17(b): A Scatternet after executing the relay reduction protocol

Although relay reduction is fine for less traffic load, suppose R(P_3, P_5) has changed its status from relay to slave because its function can be executed by R(P_1, P_3, P_5), once the traffic load increases that creates bottleneck as this single relay node is in three different piconets. In another case, the relay R(P_1, P_3, P_5) could be disconnected due to a power failure or mobility. Reducing the number of bridges also increases congestion on the active bridge nodes. The active bridges may also change their status from a bridge to a slave that has a lower degree but is the most frequently used. The data traffic load on a bridge node may also increase the delay for the connected piconets because, multiple piconets cannot communicate at the same time, therefore, they have to wait for a bridge to become free.

2.11.3 Dynamic Congestion Control through the backup relay in the Bluetooth scatternet

Subsequently, the Dynamic Congestion Control (DCC) through the backup relay in the Bluetooth scatternet (Tahir Bakhsh et al. 2011) proposes a solution for avoiding the congestion problem in the scatternet. DCC has proposed a new protocol for Congestion handling by introducing a new device called the backup relay. When a single bridge device contributes in the communication of multiple piconets, due to the congestion, it may create a bottleneck. The master device can determine the traffic load within the piconet and determine the delay from the bridge device. It maintains a relay connection table, which provides local information of the relay contribution in other connected piconets and determines the data traffic flow. When the master device senses congestion in the piconet, it activates the backup relay to share the data traffic from the same piconet. For instance in Fig.2.18, in piconet P_3, more devices are communicating through the master device, M_3, so the data traffic load can be easily determined by the master device. As different piconets are communicating through bridge device B_1, when the master device of P_3 determines the traffic load, it activates the Backup Relay (BR) to share the traffic load. DCC provides the solution for congestion handling only when more devices are communicating from the same piconet. DCC is good for intra-piconet congestion handling but it does not provide a solution for inter-piconet congestion handling. As multiple piconets are communicating through bridge device B_1, if the master device of P_3 does not find any congestion on bridge device B_1 because load on B_1 is not form P_3, then the DCC protocol fails to avoid the inter-piconet congestion.

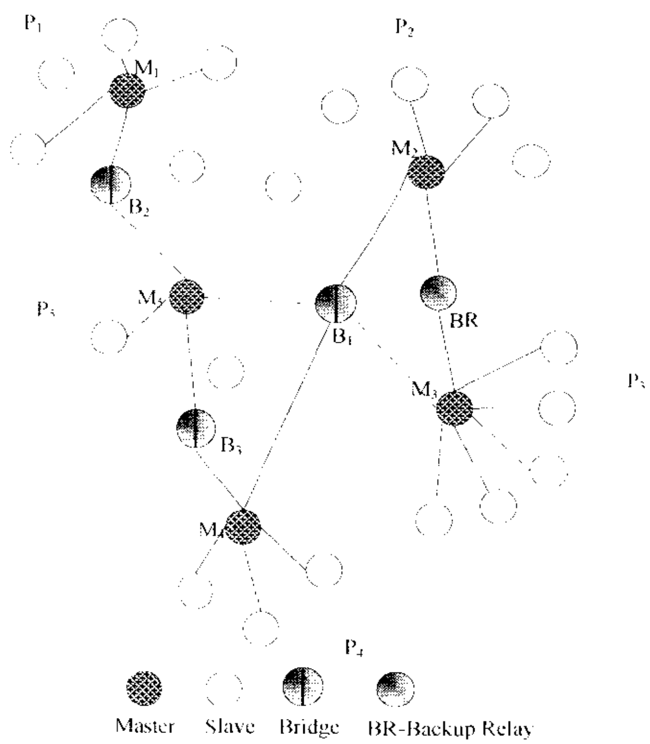


Figure 2.18: Scatternet formation using DCC Protocol

The other problem with the DCC protocol is how the DCC protocol will overcome the problem of congestion when any master device monitors the congestion within a piconet, the piconet is connected with the other piconets through a single bridge device and parallel transmissions are also required. For example in Fig.2.18, devices in (P_2 and P_5) and devices in (P_3 and P_4) want to communicate with each other through B_1 at the same time; because of having congestion, it would be difficult so first P_2 and P_5 will communicate and at the same time P_3 and P_4 will wait for B_1 to become free. In this case the DCC protocol does not allow the parallel transmissions.

2.11.4 Enhanced AODV routing protocol for Bluetooth scatternet

For Bluetooth scatternet communication, a core protocol "Enhanced AODV routing protocol for Bluetooth scatternet" was proposed (Jarrah O.M and Megdadi. 2009). This protocol enhances the main route discovery process of the Ad hoc On-Demand Distance Vector (AODV) (Al-Jarrah O and W. 2006). The Enhanced AODV protocol uses two types of algorithms; one is the traffic intensity calculation algorithm and the other is the power prediction algorithm. The traffic intensity algorithm is used for the congestion monitoring in a scatternet. The traffic intensity algorithm counts the number of received bits after every time second. So, the number of received bits incremented as a node receives the packets. For the power prediction algorithm, this protocol considers a Bluetooth device in one of the mentioned modes: Idle, Send and Receive. A Bluetooth device consumes energy when it transmits or receives data packets or control packets. It also consumes energy even in the idle state. For energy consumption, this protocol uses the bandwidth functions that release the node battery power depending on the existing device state and the transmitted/received packet size. It finds the main route by flooding the route request packets to all the devices (slave, master and bridge) in the scatternet. This protocol keeps all the incoming duplicated route request packets for the purpose of the best route selection. The route request packet of this protocol consists of the average traffic intensity, predicted power, expected reply time and Bluetooth ID list. This protocol floods the route request packets to all the devices (masters, slaves and bridges) in the network. As for the main route construction for any pair of source and destination, the routing link contains the intermediate master and bridge devices. So, when it floods the route request packets in all of the intermediate piconets, the slave devices consume extra energy and time. Another problem with this protocol is that it chooses the main route from the available routes but it does not mention any efficient protocol for route optimization. In Fig.2.19, suppose device B wants to communicate with the device J. It starts searching for the route by flooding the route request packets in the whole network and waits to get all of the possible routes. The problem with the longest route is, as it contains multi-hops and more intermediate devices, it consumes extra resources and time. Unnecessary intermediate devices sometimes are predicted as weak and can also make the main link weak, which in turn can create more data loss.

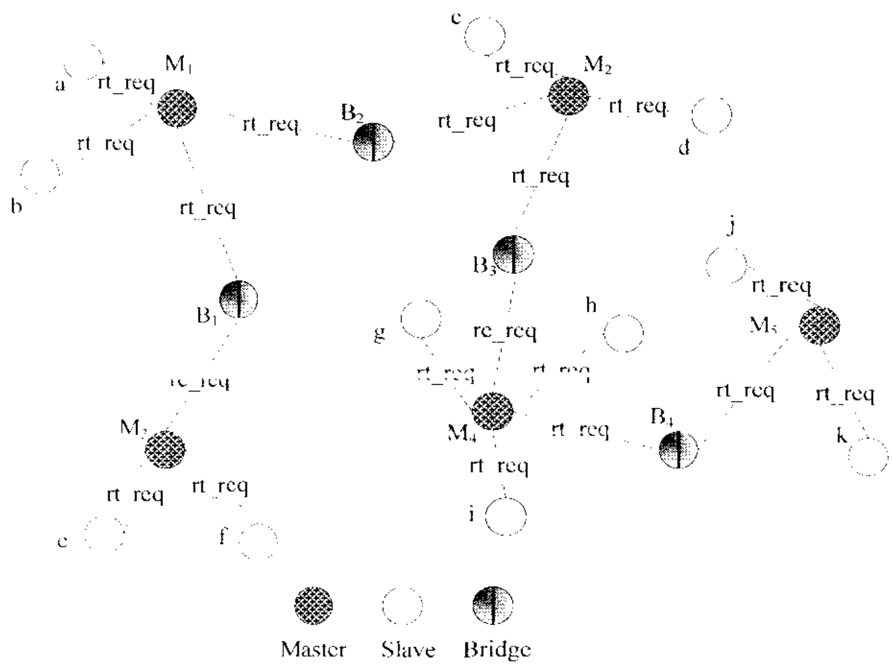


Figure 2.19: Finding the main route by using the Enhanced AODV protocol

2.11.5 Dynamic energy aware network maintenance for Bluetooth

In Dynamic Energy aware Network Maintenance for Bluetooth (DENM) protocol (Bakhsh et al. 2011), a master device maintains a table where it keeps the record of all connected slave devices. When the energy level of a master device reaches L_1, it activates an Auxiliary Master (AxM) from its slave devices. In the same way, if the energy level of a bridge device reaches L_1, it sends a request message to the connected master devices for backup relay activation. This protocol provides flexibility against frequent link disconnection based on energy level. The work of the DENM protocol is explained by a diagram. In Fig.2.20 (a), source device A transmits data to destination device E through the (A-M_1-B_1-M_2-E) route. After some time, the energy level of M_1 starts decreasing and reaches the limit. In Fig.2.20 (b), as M_1 has the list of backup devices, it activates F as a new master and starts transmission through the A-F-B_1-M_2-E route.

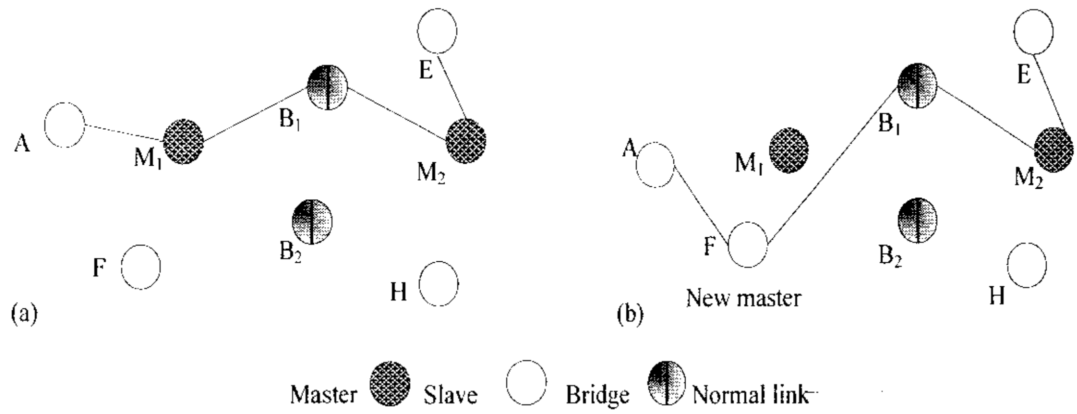


Figure 2.20: Route maintenance by using DENM protocol

The drawback with this protocol is that if the devices frequently communicate with each other, then they will lose their energy very soon and break the links. So, on the basis of energy level only, it cannot be determined as whether the device has moved away or the link has broken.

2.11.6 Bluetree

A Bluetree (Tekkalmaz M et al. 2006) creates a tree like structure. In order to decrease the congestion and make the energy consumption of devices more efficient, a Bluetree transfers the devices from their origional positions. To overcome the traffic load problem of the Bluetree, another protocol "An efficient reconstruction approach for improving Bluetree scatternet formation in personal area networks" was proposed (Lin J.W, et al, 2010). This protocol utilises the Bluetree model where it transfers the whole piconet instead of just transferring a single node. The piconet movement uses tree traversal procedures (post order traversal and level order traversal).The post order traversal is used to visit each piconet from the bottom to the top. Whereas, the level order traversal approach visits each piconet from the top to the bottom.The tree traversal procedure is used to monitor each piconet to findout its best location. There is only one piconet on the top of a Bluetree scatternet called the root piconet. Other piconets are given their level numbers according to their distance from the root piconet. The piconet based Bluetree scatternet formation is explained by an example in Fig.2.21 (Lin J.W et al. 2010). Suppose node *a* from piconet P_1 and node *d* from piconet P_4 are communicating. The link is passing through piconet P_2. On the other side, node *g* from piconet P_6 and node *i* from piconet P_7 are communicating. The routing link is passing through piconet P_3.

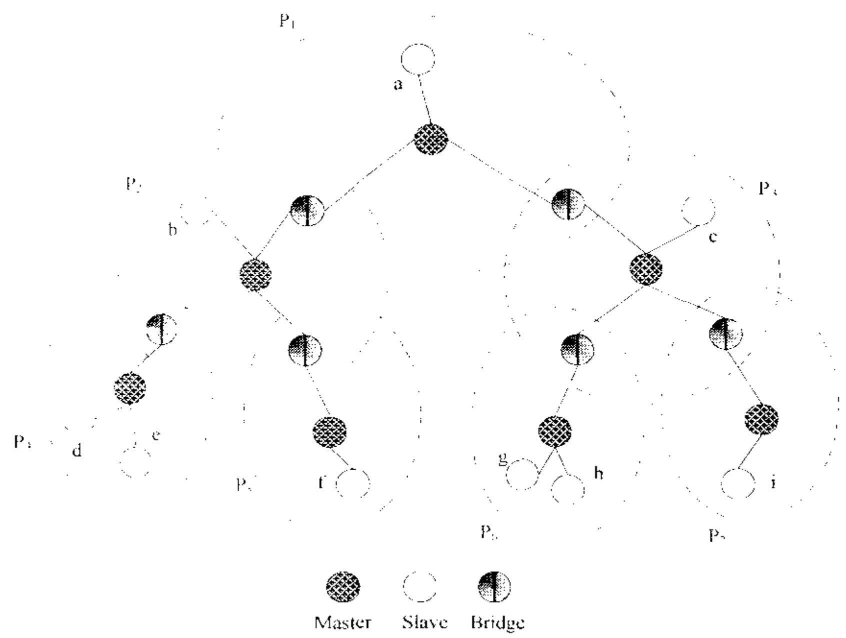


Figure 2.21: Piconet based Bluetree scatternet

Fig.2.22 shows the scatternet traversing and piconet movement, where each piconet is represented by a circle. Fig.2.22 (a) represents the piconets by single nodes(circles). Fig.2.22(b) shows how the piconets move and change their position. As P_1 and P_4 are communicating, it moves P_4 upwards, and piconets P_6 and P_7 are communicating so it moves P_7 downwards.

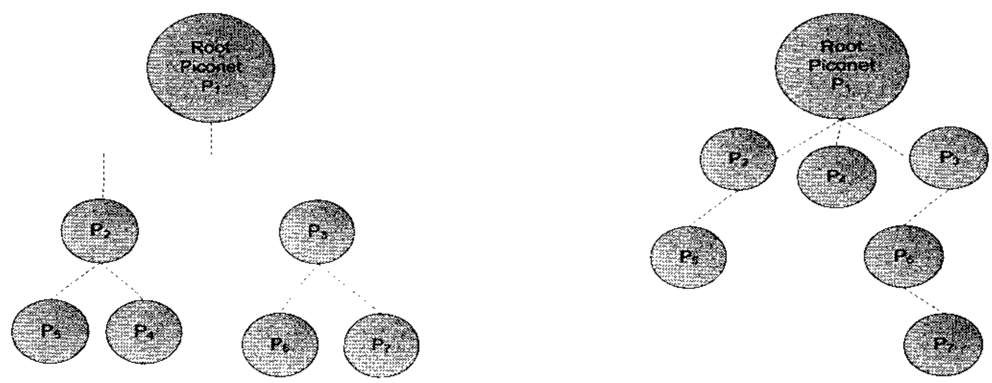


Figure 2.22(a): Piconet Bluetree scatternet (b): Piconet movement

The problems with this protocol are that as it moves the whole piconet, thus it increases control overhead. In addition, as it traverses the network for communication between piconets, which increases delay. It also increases probability of route breakage; in case devices from P_5 and P_4 start communication at that time when P_4 starts moving upwards during this situation, it can break the link between P_4 and P_5.

2.11.7 A Novel route maintenance protocols for the Bluetooth ad hoc network with mobility

The other relevant protocol is the Novel route maintenance protocols for the Bluetooth ad hoc network with mobility (ROMA) (Sahoo et al. 2008). Firstly, a routing path of Bluetooth scatternet is maintained with ROMA. If devices are added or removed from the network, ROMA decreases the number of hops by adding new devices and reconstructs the routing path. ROMA follows the following procedures for route maintenance:

a) Algorithm for Node Add Procedure: In the routing path of the Bluetooth scatternet, if a new slave device enters into the network, the respective master device verifies whether this device can reduce the routing path or not. For the device verification, a respective master device executes the route optimisation algorithm.
b) Algorithm for device leaving procedure: Within the routing path of the Bluetooth scatternet, if a device moves away from its original position, the

device leaving algorithm is used. This algorithm contains the sub procedures for the route maintenance.

i. Device Replacement Procedure: This procedure is used to find out the device to replace the damaged device.

ii. Link Replacement Procedure: This procedure is used to reconstruct the sub route by replacing the link.

In the ROMA protocol, if a new device joins or leaves the scatternet it maintains its routing link by considering the number of the hop count. The respective master device checks the new device as to whether it can reduce the number of the hop count and make the routing path shorter; then it changes the role of the new device and creates new piconet(s) within the routing path. Although this protocol works well, but it does not consider the changes of device's role once the communication has ended. Moreover, ROMA does not provide any solution in the routing path for the devices that want to communicate with their original master device or slave device in the network. According to the ROMA protocol, if a Bluetooth device moves away from the piconet it informs its master device for the device leaving procedure which takes extra time and consumes more energy. In addition, the ROMA protocol does not consider how it will inform to the routing master if a device suddenly fails or goes out from the radio range of routing master. Once the link breaks or device fails, the ROMA protocol finds another device to repair the broken link; in the case where no such device exists that can connect the path then it will take more time to find out another way for connecting the broken links.

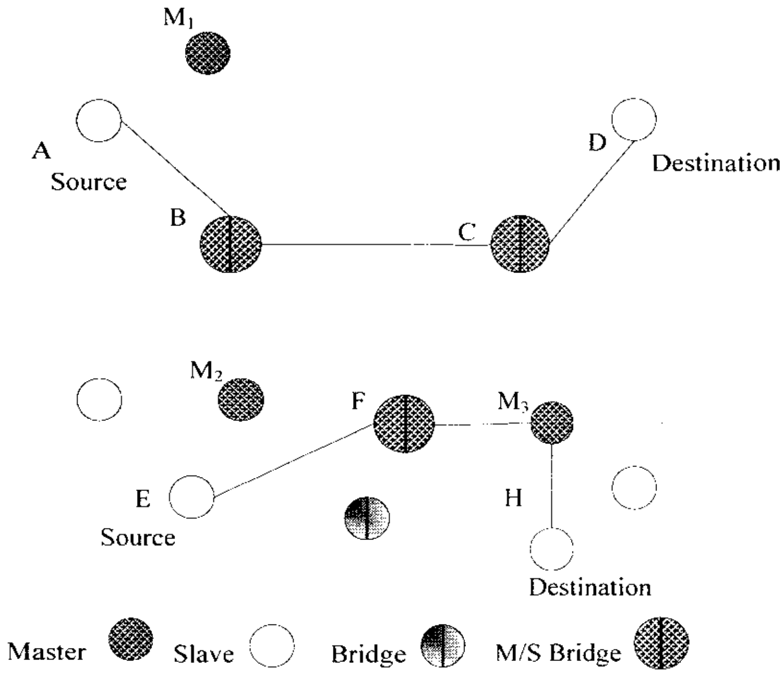


Figure 2.23(a): Routing in a scatternet through the ROMA protocol

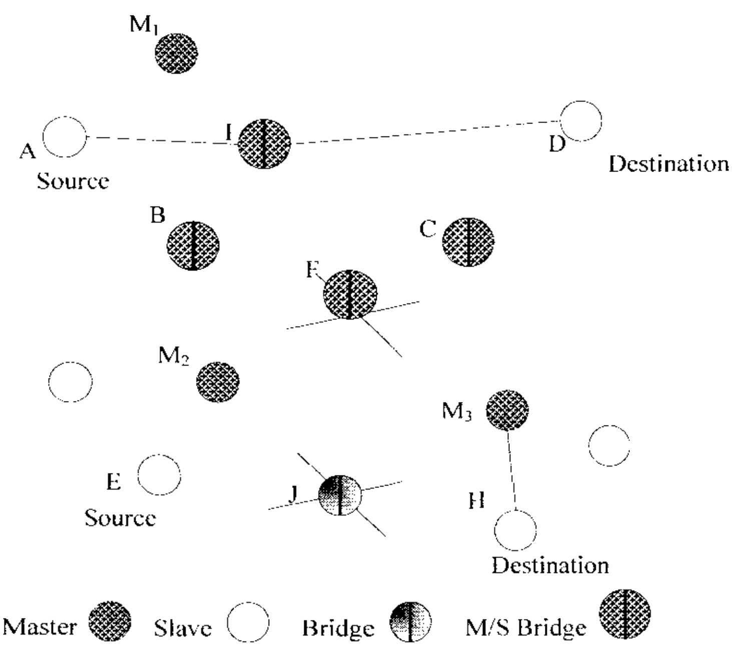


Figure 2.23(b): Problems in the ROMA protocol

As an example shown in Fig.2.23 (a), source device A is transmitting data to destination device d through the (A-B-C-D) route and data is also routed from source device E to H through the routing path (E-F-M_3-H). To reduce the number of hop counts, the ROMA protocol changes the status of B, C and F from a slave to a master-slave (M/S) bridge. According to the Bluetooth rules, one bridge device can be master for only one piconet but it can be slave for more piconets.

Fig.2.23 (b) shows that during data transmission, a new device (I) enters into the piconet of M_1. As M_1 is the routing master, it checks that this device can provide the shortest path. As devices I and D are directly within the radio range of each other, the routing master breaks the existing link and makes a new link (A-I-D). Meanwhile, device F starts moving and goes out from the range of routing masters M_2 and M_3. When it starts moving, it informs the routing masters so they can find another device, J, which can re-establish the route (E-J-M_3-H). Device E is now transmitting data through a new route but suddenly device J fails and it could not inform its routing master. During this situation, the ROMA protocol does not provide any solution to overcome the problem.

The problems in the existing works provide an opportunity to propose a new protocol for efficient scatternet communication to overcome the inefficiencies of existing protocols. Although, many researchers have proposed different ideas for Bluetooth network but in this research, the authors have considered only the most relevant protocols.

2.12 SUMMARY

This chapter has completely illustrated the literature review. The first part of this chapter described the full overview of the Bluetooth technology. Furthermore, the Bluetooth protocol stack, device states and packet structure is discussed. This chapter also categorized the Bluetooth network topologies as different Bluetooth protocols use different scatternet topologies. The applications of the Bluetooth technology demonstrate its usage in different organizations. The Bluetooth short range communication is more beneficial because it helps the devices to connect with each other and exchange data easily. Finally, the last part described the explanation of the existing protocols and it also pointed out the shortcomings in the existing protocols.

CHAPTER 3

METHODOLOGY

The literature review was presented in Chapter 2 where the research issues were also identified. In this chapter, on the basis of research issues, the methodology is included in this research work that briefly discussed the congestion handling, route maintenance and a new combined protocol of congestion and route maintenance. Finally, this chapter concludes the methodology.

Proposed Protocols

In this part the first section based on network congestion handling and the second section is related to the route maintenance of the dynamic Bluetooth networks. Whereas the last section contains a combined protocol, that is combination of both the congestion handling and route maintenance protocols.

3.1 AN EFFICIENT CONGESTION HANDLING PROTOCOL FOR BLUETOOTH SCATTERNET (ECHP)

This sub-section discusses some basic definitions and the ECHP protocol for intra-piconet and inter-piconet congestion handling. The protocol consists of two scenarios; in the first scenario, role switching techniques were used to overcome the problem of intra-piconet congestion handling which had two cases Piconet Formation within a Piconet (PFP) and Scatternet Formation within a Piconet (SFP). In the second scenario, a Fall-Back Bridge (FBB) was used to handle the traffic load on the bridge device that overcomes the problem of congestion in the inter-piconet.

3.1.1 System model for ECHP

The system model including the proposed system and an explanation and the symbols related to the proposed work are as follows:

3.1.1.1 Traffic Flow Table (TFT)

The proposed protocol maintains two types of TFT, one for the master node and the other for bridge node called the Master Traffic Flow Table (MTFT) and the Bridge Traffic Flow Table (BTFT). To avoid congestion on the master node through the role switching operation, the MTFT is maintained that keeps the information of all of the incoming and outgoing data traffic going through the master node within the piconet. Congestion can be easily determined on the master node by the MTFT; while, each bridge node maintains a BTFT that keeps the information of all of the connected masters. This table shows the information of all of the communicating links going through the bridge node, thus, the bridge node can detect congestion for the inter-piconet through the BTFT.

3.1.1.2 Node Information Table (NIT)

Each master node maintains a Node Information Table (NIT), which contains the node ID, Clock-offset etc.

3.1.1.3 Most Frequently used Nodes (MFN)

During transmission, some devices contain heavy data to share with the other devices therefore they transmit data again and again (frequently) until the complete data exchange. The proposed ECHP protocol performs the role switching operation only on the MFN(s) to overcome the congestion problem.

3.1.1.4 Auxiliary Master (AxM) and Auxiliary Bridge (AxB)

An Auxiliary Master (AxM) is a slave node in an original piconet but becomes the master node in the new piconet, whereas any intermediate node between the source and destination can be selected as an Auxiliary Bridge (AxB). During the role switch operation, the proposed protocol selects a source node as an AxM; if any intermediate node is required it would be selected as an AxB.

3.1.1.5 Park Mode (PM)

During the role switch operation, the proposed protocol uses the park mode for those nodes which will change their status in new piconets. The purpose of using the park mode is saving active member addresses.

3.1.1.6 Threshold

The threshold value is used for congestion handling on the master and bridge nodes. When MF communicating nodes reach the limit of the threshold value, the proposed protocol performs network restructuring.

3.1.1.7 Received Data

The incoming data is called the Received data (*Rx*), where the master device of a piconet records each data packet it sends to any device in the piconet. The first subscript (*1*) in a_{1j} corresponds to the first row, that master records all incoming data; the second subscript (*j*) corresponds to the column. Whereas, *n* represents total number of columns.

The Rx is calculated by the following equation 3.1:

$$Rx = \sum_{j=1}^{n} a_{1j} \quad for\, j = 1, 2, 3, \dots\dots\dots n\; where\; n < 8 \tag{3.1}$$

3.1.1.8 *Transmit Data*

The outgoing data is called the Transmit data (*Tx*), where the master device of a piconet records each data packet, when any device sends a data packet within or outside the piconet. The first subscript (*i*) in a_{i1} corresponds to row, that master records all outgoing data; the second subscript (*1*) corresponds to the first column. Whereas, *m* represents total number of rows.

The *Tx* is calculated by the following equation 3.2:

$$Tx = \sum_{i=1}^{m} a_{i1} \quad for\ i = 1, 2, 3, \dots \dots \dots m\ where\ m < 8 \tag{3.2}$$

3.1.1.9 *Total Data Flow*

The sum of the received and transmitted data is the Total Data Flow (TDF) that can be calculated by the following equation 3.3, where *i* refers the row and *j* refers the column.

$$Rx + Tx = \sum_{i=1}^{n} a_{i1} + \sum_{j=1}^{m} a_{1j} \tag{3.3}$$

3.1.2 Analytical model for the connected congested scatternet

Suppose *N* is the total number of Bluetooth devices (slaves, masters and bridges) in a scatternet as in equation 3.4.

$$N = S \cup M \cup B \tag{3.4}$$

where *S* denotes the number of slaves, *M* denotes the number of masters and *B* denotes the number of bridges. The master device connects the maximum of 7 devices within its piconet (*P*) and the maximum distance between the master and slave devices is 10m.

$$M = \sum_{i=0}^{s} \quad \text{where } s = \{1, 2, \ldots \ldots 7\} \tag{3.5}$$

If $P_i \longrightarrow \{(m_i$ is the master of piconet (P_i) and $S_{i,j}$ indicates the slave with the maximum distance of 10m, where P_i is the i^{th} piconet and $S_{i,j}$ is j^{th} slave in i^{th} piconet.

A connected scatternet has intermediate devices (bridge/relay) to provide communication in multiple piconets. Equation 3.6 represents an intermediate device relay, where P_i and P_j are any two piconets and n_b is an intermediate device between these piconets.

$$\textit{Suppose } P_i \,\&\, P_j \textit{ are connected} \longrightarrow \textbf{then } P_i \cap P_j = n_b \quad \text{where } i \neq j \tag{3.6}$$

The Congestion on a Master device (*CM*) can be calculated by analyzing the outgoing and incoming traffic as follows in equation 3.7, where i and j correspond to row and columns respectively.

$$CM = \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} \; for \; i \,\&\, j = 0,1,2 \ldots 7 \tag{3.7}$$

As the bridge device receives/transmits data from the master devices, if the bridge device receives the data from the master device it is considered as Bridge Received data, *BRx*; similarly, if the bridge device transmits data to the master device it is considered as the Bridge Transmit data, *BTx*.

The Congestion on a Bridge device *CB* can be calculated by the following equation 3.8, where i and j corresponds to the row and columns respectively.

$$CB = BRx + BTx = \sum_{i=1}^{n} a_{i1} + \sum_{j=1}^{m} a_{1j} \; for \; i \,\&\, j = 0, 1, ..7 \tag{3.8}$$

3.1.3 Intra-piconet congestion handling through the dynamic role switching operation

In this section, intra-piconet congestion handling is presented; the intra-piconet congestion is handled through the PFP and SFP.

3.1.3.1 Intra-piconet congestion handling through the Piconet Formation within a Piconet (PFP)

Passing large numbers of connections through the master node within a piconet may create the congestion. As a master is always involved with all incoming and outgoing data traffic, it can easily determine the data traffic load. When a master node determines the higher traffic load, it performs the role switch operation by sending a request packet to the pair of MFN within the piconet. According to the proposed protocol, initially all master nodes maintain a Traffic Flow Table (TFT). When a master node encounters the congestion in the piconet, it sends the role switch request to the pair of source and destination nodes. When the source and destination nodes receive the master node's request, both nodes change their status. In the next step, the master node changes its mode (into park mode). Once the communication ends, the nodes come back into the original states, and send requests to re-join the master node. As the master node maintains the list of nodes for temporary connections thus, the original active member addresses are reserved by the master node. Hence, the nodes can come back to the original state without losing their connections.

3.1.3.2 Analytical analysis of PFP

As shown in Fig.3.1 (a), nodes (i, j) and (a, b) are most frequently communicating pairs so M_1 and M_2 perform the piconet restructuring. Therefore, a request is sent to the slave nodes by the master nodes. As a result, after the piconet restructuring, the slave nodes (j and a) became auxiliary masters. The new connections of the Most Frequently MF communicating nodes are shown in Fig.3.1 (b). Table 3.1 explains the data flow (receive and transmit data) of M_2, where 1 represents the link between nodes and Ø represents that there is no connection between nodes. The MF represents that the link is most frequently used link.

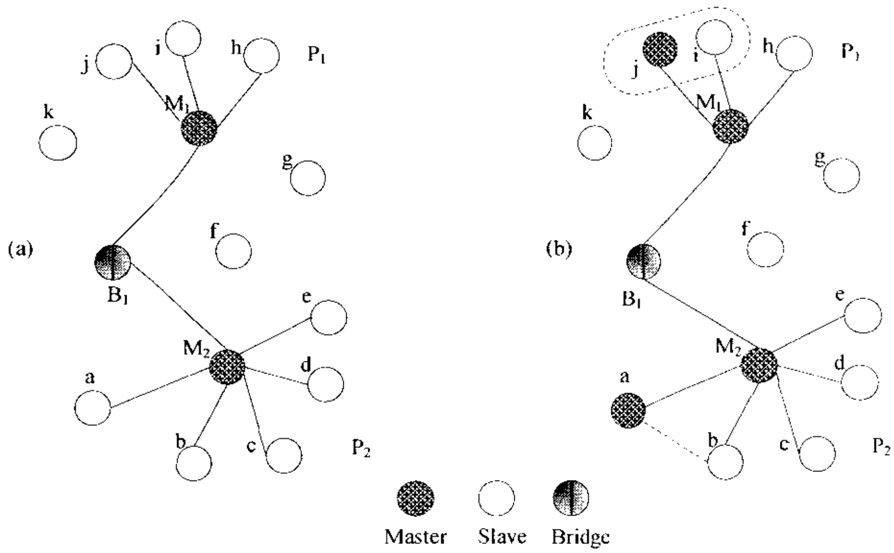


Figure 3.1(a): Before the role switching operation (b): After the role switching operation

Algorithm 1: Piconet Formation within a Piconet

Notations:	**Functions:**
N: total number of nodes n: node Tr_ld: Traffic load Src: Source node Dst: Destination node AxM: Auxiliary Master MF: Most frequently communication pair	Disconnect ( ): Break link Req_role_swch ( ): Request for role switch operation Park_mode ( ): Assign park mode Inq ( ): Inquiry Inq_scn ( ): Inquiry scan Pg ( ): Page mode Pg_scn ( ): Page scan mode Update_NIT ( ): Update node information table Role ( ): Node role Start_comm ( ): Start communication

Input: *Node Information Table*

Output: *Create direct link between most frequent communicating devices*

Begin

1. ***foreach*** $n_i \in N$ ***do***
    1.1. ***if*** is_slave (n_i) = true ***then***
        1.1.1. Inq_scn (n_i)
    1.2. ***else***
        1.2.1. Inq (n_i)
        end if
    endforeach

2. ***foreach*** $n_i \in N$ ***do***
   2.1. ***if*** is_master (n_i) = true ***then***
      2.1.1. Pg (n_i)
      2.1.2. Update_NIT (n_i)
   2.2. ***else***
      2.2.1. Pg_scn (n_i)

      end if

   endforeach

3. ***foreach*** $n_i \in M$ ***do***
   3.1. ***if*** Tr_Id < Ø ***then***
      3.1.1. Start_comm ( )
   3.2. ***else***
      3.2.1. select MF
         3.2.1.1. ***If*** node = Src ***then***
            3.2.1.1.1. Req_role_swch (AxM)
            3.2.1.1.2. Park_mod (Src)
            3.2.1.1.3. Park_mod (Dst)
               end if

         end if

      endforeach

4. ***do***
   Transmission

   While (Communication)

5. disconnect (Src & Dst)
   5.1. Req_role_swch (Src)
   5.2. Req_role_swch (Dst)

End

Table 3.1: Data traffic flow analysis on the master M_2

ID	a	b	c	d	e	f	B_1	M_2
a	Ø	MF	Ø	Ø	Ø	Ø	Ø	1
b	MF	Ø	1	Ø	Ø	Ø	Ø	1
c	Ø	Ø	Ø	1	1	Ø	Ø	1
d	Ø	Ø	1	Ø	1	Ø	Ø	1
e	Ø	Ø	1	1	Ø	Ø	Ø	1
f	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
B_1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
M_2	1	1	1	1	1	Ø	1	Ø

Table 3.2: Node Information Table NIT for Piconet P_2

ID	Clock-offset	Device-role	Received data	Transmit data
a	C-offset (a)	AxM	500	600
b	C-offset (b)	S	600	500
M_2	C-offset (M_2)	M	1100	1100

The concerned steps of Piconet Formation within the Piconet PFP are as follow:

1. In the first step, the master node enters into the inquiry state and the slave nodes enter into the inquiry scan state.
2. In the second step, the master node performs paging and the slave nodes perform page scan operation for synchronisation. Finally, all of the active Bluetooth nodes are in their original piconets and communicating through their master nodes.
3. In the third step, the master node sets a threshold value for monitoring the congestion. The master node maintains the NIT and monitors the congestion by the threshold value; if it finds congestion within the piconet:
   a) Master node selects MF communicating pair of the source and destination slave nodes.
   b) Master node sends a request packet of the role switching operation to the pair of the source and destination.
   c) Master node updates the record of the source and destination as in the park mode.
   d) Source node is selected as an auxiliary master.
4. Once the new piconet is constructed, the Bluetooth nodes can directly communicate.
5. Finally, when the communication ends successfully, all Bluetooth nodes return to their original states.

3.1.4 Intra-piconet congestion handling through the Scatternet Formation within a Piconet (SFP)

When a master node senses the congestion and finds that the MF communicating pair of the source and destination nodes is not directly within the radio range of each other, then it performs the role switch operation by making a scatternet within a piconet. During this operation, an intermediate node is selected as an auxiliary bridge and a pair of source and destination nodes is selected as an auxiliary master.

3.1.4.1 Analytical Analysis of SFP

The SFP operation is explained through Fig.3.2. As nodes (h) and (f) are MF communicating nodes in the same piconet, but both are not within the direct radio range of each other, thus, node (g) is selected as an AxB and node (h) and (f) are AxMs. The Table 3.3 shows the data traffic flow analysis on the master node M_1. where 1 represents the link between nodes and $\emptyset$ represents that there is no connection between nodes. The MF represents that the link is most frequently used link. In Table 3.4, according to Fig.3.2, the master node of P_2 updates the NIT for the MF communicating nodes which are in the same piconets but cannot communicate directly so they need an intermediate node.

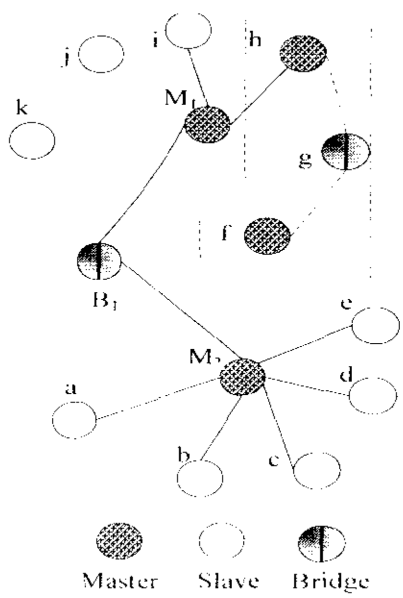


Figure 3.2: handling of congestion by making a scatternet within the piconet

Algorithm 2: Scatternet Formation within the Piconet, SFP

Notations:	**Functions :**
N: Total number of nodes	Inq ( ): Inquiry
n_i: node	Inq_scn ( ): Inquiry scan
Src: Source	Pg ( ): Page mode
Dst: Destination	Pg_scn ( ): Page scan mode
AxM: Auxiliary Master	Req_role_swch ( ): Request for role switch operation
AxB: Auxiliary Bridge	Park_mod ( ): Assign park mode
Tr_ld: Traffic load	Disconnect ( ): Break link
	Update_NIT ( ): Update node info table

Input: *Node Information Table*

Output: *Activate the new route between frequent communication devices*

Began

1. ***foreach*** $n_i \in N$ ***do***
   1.1. ***if*** is_slave (n_i) = true ***then***
      1.1.1. Inq_scn (n_i)
   1.2. ***else***
      1.2.1. Inq (n_i)
      end if
   endforeach
2. ***foreach*** $n_i \in N$ ***do***
   2.1. ***if*** is_master (n_i) = true ***then***
      2.1.1. Pg (n_i)
      2.1.2. Update_NIT (n_i)
   2.2. ***else***
      2.2.1. Pg_scn (n_i)
      end if
   endforeach
3. ***foreach*** $n_i \in M$ ***do***
   3.1. ***if*** Tr_ld < Ø ***then***
      3.1.1. Start_comm (Src, Dst)
   3.2. ***else***
      3.2.1. select MF
         3.2.1.1. ***if*** nodes can communicate directly ***then***
            3.2.1.1.1. Req_role_swch (Src)
            3.2.1.1.2. Req_role_swch (Dst)
         3.2.1.2. ***else***
            3.2.1.2.1. select intermediate node
            3.2.1.2.2. role = AxB,
            3.2.1.2.3. Inform source and destination about AxB
            3.2.1.2.4. ***if*** node = Src ***then***
               3.2.1.2.4.1. role = AxM
               3.2.1.2.4.2. Park_mod (AxM)
               3.2.1.2.4.3. Park_mod(AxB)

3.2.1.2.4.4. Park_mod(Dst)

end

3.2.1.3. Req_role_swch (Src)
3.2.1.4. Req_role_swch (Dst)
3.2.1.5. Req_role_swch (AxB)

end if

end if

endforeach

4. ***do***
   4.1. Transmission
   4.2. ***While*** (communication)
5. Disconnect (AxB)
   5.1. Req_role_swch (Src)
   5.2. Req_role_swch (Dst)
   5.3. Req_role_swch (AxB)

End

Table 3.3: Data traffic flow analysis on the master node, M_1

ID	f	g	h	i	j	k	B_1	M_1
f	Ø	1	MF	1	1	1	1	1
g	1	Ø	1	1	1	1	1	1
h	MF	1	Ø	1	1	1	1	1
i	1	1	1	Ø	1	1	1	1
j	1	1	1	1	Ø	1	1	1
k	1	1	1	1	1	Ø	1	1
B_1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1
M_1	1	Ø	Ø	1	1	1	1	Ø

Table 3.4: Node Information Table NIT for P_1 after SFP

ID	Clock-offset	Device-role	Received data	Transmit data
f	C-offset (f)	AxM	700	800
g	C-offset (g)	AxB	1500	1500
h	C-offset (h)	AxM	800	700

The concerned steps of the Scatternet Formation within a Piconet, SFP:

1. In the beginning, the master node enters into the inquiry state and the slave nodes enter into the inquiry scan state.
2. In the second step, the master node performs the paging and the slave nodes perform the page scan for the synchronisation.
3. When a piconet is constructed, then the devices may communicate with each other.
   3.1 The master node monitors the threshold value, if it exceeds the threshold. Then the master node checks the MF communicating pair of the source and destination if the nodes are not in the direct radio range of each other in the same piconet.
   3.2 The master node checks an intermediate node between the source and destination.
      a. The master node sends a request packet of the role switching operation to the intermediate node and the pair of the source and destination.
      b. The master node selects an intermediate node as the AxB in the same piconet.
      c. The source and destination of both of the nodes are selected as AxMs.
      d. The master node updates the record of the AxB and the pair of AxMs as in the park mode.
4. Now an efficient scatternet within a piconet is created and the nodes are communicating.
5. Once the communication ends successfully, the nodes return into their original states.

3.1.5 Inter-piconet congestion handling on the bridge node through a Fall-Back Bridge (FBB)

It has been analyzed that when a single bridge node participates in multiple piconets, it causes inter-piconet congestion. Inter-piconet congestion mostly occurs due to the unavailability of a bridge node. Therefore, the proposed protocol solves the inter-piconet congestion problem through the FBB. According to the proposed protocol, each active bridge node maintains a Bridge Traffic Flow Table (BTFT) that contains the information of all of the connected masters and the traffic load that has passed through the bridge node. If a bridge node detects congestion, it transmits a request packet to the connected master nodes for accumulating the information of their bridge nodes. On receiving the bridge request, each master searches in the NIT for a FBB; if any master node has a FBB, then it sends a reply to the bridge node to activate its connection with the required master node. Meanwhile, the master node sends the active bridge node into the park mode.

3.1.5.1 Analytical Analysis of FBB

As shown in Fig.3.3 (a), multiple piconets are communicating through B_1, and they create congestion on B_1 due to the heavy data traffic load. The heavy traffic flow does not allow the parallel transmission in the scatternet, thus, B_1 creates a bottleneck. As one master node sends data through B_1, at the same time others wait for B_1 to become free.

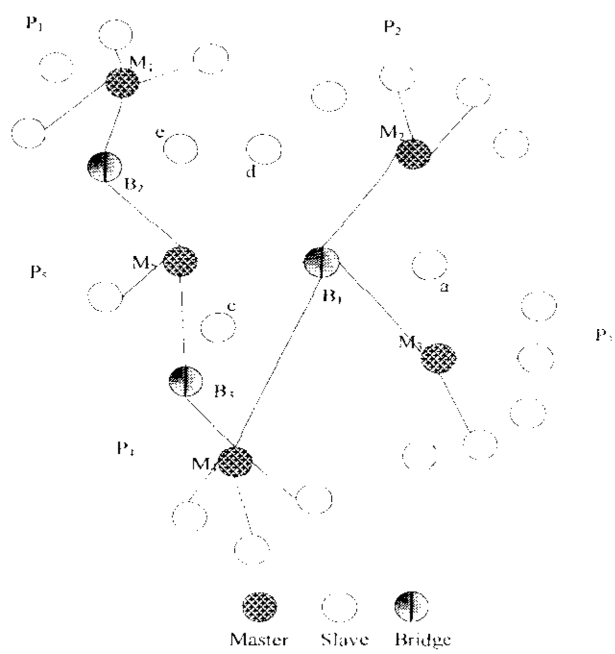


Figure 3.3(a): Scatternet communication before using the FBB

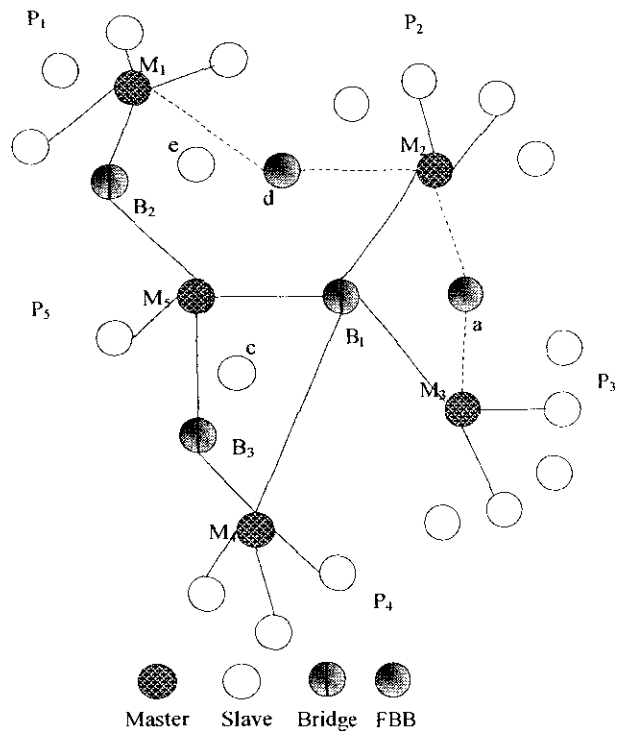


Figure 3.3(b): Scatternet communication after using the FBB

The FBB concept is used when: a single bridge node is not sufficient for an efficient communication between piconets and when parallel transmissions are required between the piconets for well organized and smooth communication. In Fig.3.3 (b), different piconets are communicating through B_1 and due to the congestion, node (a) is selected as the FBB for P_2 and P_3 and node (d) is selected as the FBB for P_1 and P_2. When multiple bridges are available a higher degree node is selected as the FBB. The purpose of selecting higher degree bridge node (as the FBB) is for providing more connections among the different piconets. As shown in Fig.3.3 (b), the dotted lines show the temporary links where the traffic is shared and the bottleneck is resolved. When the master nodes update their status for the FBB, they set the FBB in park mode. After successful communication, the FBBs return to their original states.

Algorithm 3: Inter-piconet congestion control through a Fall-Back Bridge

Notations:	**Functions:**
N: Total number of nodes	Inq ( ): Inquiry
n: Node	Inq_scn ( ): Inquiry scan
Tr_ld: Traffic load	Pg ( ): Page mode
Src: Source	Pg_scn ( ): Page scan mode
Dst: Destination	Park_mod ( ): Assign park mode
FBB: Fall-Back Bridge	Connect ( ): Create link
	Disconnect ( ): Break link
	Update_NIT ( ): Update node info table

Input: *Bridge Traffic flow Table*
Output: *Activates the Fall-Back Bridge*
Begin
1. ***foreach*** $n_i \in N$ ***do***
    1.1. ***if*** is_slave (n_i) = true ***then***
        1.1.1. Inq_scn (n_i)
    1.2. ***else***
        1.2.1. Inq (n_i)
        end if
    endforeach
2. ***foreach*** $n_i \in N$ ***do***
    2.1. ***if*** is_master (n_i) = true ***then***
        2.1.1. Pg (n_i)
        2.1.2. Update_NIT (n_i)
    2.2. ***else***
        2.2.1. Pg_scn (n_i)
        end if
    endforeach
        2.2.2. ***foreach*** $n_i \in M$ ***do***
    2.3. ***if*** Tr_ld < Ø ***then***
        2.3.1. Start_comm ( )
        2.3.2. ***else***
            2.3.2.1. Request all connected masters
            2.3.2.2. select FBB
3. FBB reply to connected masters
    3.1. Update Table
4. ***if*** role = FBB ***then***
    4.1. Req_role_swchsend (FBB)
    4.2. park_mod (FBB)
    4.3. Connect (FBB)
        end if
5. **do**
    5.1. Transmission
    5.2. ***while*** (communication)
6. Disconnect (FBB)
End

Table 3.5: Bridge Traffic Flow Table for B_1

ID	M_2	M_3	M_4	M_5	B_1
M_2	Ø	MF	1	1	1
M_3	MF	1	1	1	1
M_4	1	1	Ø	1	1
M_5	1	1	1	Ø	1
B_1	1	1	1	1	Ø

The steps for congestion control on the bridge node through the Fall-Back Bridge are as follows:

1. Firstly, the master node enters into the inquiry state and the slave nodes enter into the inquiry scan state.
2. In the second step, the master node performs the paging and the slave nodes perform page scan for the synchronisation.
3. Now, all of the active piconets are communicating through bridge nodes. Where the bridge node maintains a BTFT. The bridge node sets a threshold value for monitoring the congestion. If the value of the data traffic exceeds the value of the threshold, then the bridge node transmits a request packet to the connected master nodes for the activation of the FBB.
4. In this step, all of the connected masters check the intermediate node that can be used as the FBB. The master nodes having the FBB replies to the bridge node. The master nodes update their status. The master nodes transmit a request packet to the intermediate node to become the FBB and go into the park mode.
5. The FBB makes temporary links between the piconets. The piconets can communicate through the FBB.
6. After a successful communication, the temporary links are disconnected and the FBB returns to its original state.

3.2 AN EFFICIENT ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC BLUETOOTH NETWORKS (ERMP)

In this sub-section, An Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks (ERMP) is proposed that predicts the route breakage before it breaks. In a scatternet, when any Bluetooth device needs to communicate with the other Bluetooth devices it first creates a transmission link between the source and destination. As mentioned in the previous section, the Bluetooth technology does not allow direct slave to slave or master to master communication so, whenever, the Bluetooth devices need to communicate with each other, they must follow the communication rules (slave-master) and (master-bridge-master). In the proposed ERMP, once the main link is established each routing master and routing bridge updates their tables and they must keep the information of the available Fall Back Devices (FBDs). During transmission, if a weak link or weak device is predicted, the proposed protocol makes a new link and overcomes the frequent link disconnection problem by using the FBDs. In case there is no FBD available, then the proposed protocol performs a role switch operation for route maintenance.

3.2.1 System model of ERMP

The system model of ERMP consists of the following basic concepts and definitions which were used in the proposed protocol.

3.2.1.1 Bluetooth Routing devices

Whenever a Bluetooth device wants to communicate with another Bluetooth device, first of all, a communication link is established known as a routing link and the intermediate devices which are used to make the routing link are called routing devices. These devices can be slave, master or bridge.

3.2.1.2 *Bluetooth Routing Inter-Piconet*

Whenever a routing link is established within a scatternet, it may pass through multiple piconets; therefore, the routing link consists of several routing master devices. So, a piconet that holds the routing master device is known as the Bluetooth Routing Inter-Piconet. Whenever a new device enters into the routing piconet, it must send its complete information to the routing master device.

3.2.1.3 *Threshold for Signal-to-Noise Ratio (SNR)*

The ratio of the signal power to the noise power is called as the Signal-to-Noise Ratio and it is abbreviated as SNR (Wei-Yi et al. 2010). Fig.3.4 shows that the received power signal strength can be utilized to estimate distance as all of the electromagnetic waves show the inverse square relationship between the distance and the received power. The equation 3.9 is as below:

$$P_r \propto 1/d^2 \tag{3.9}$$

The received power between the transmitter and receiver is indicated by Pr and d is considered as the distance.

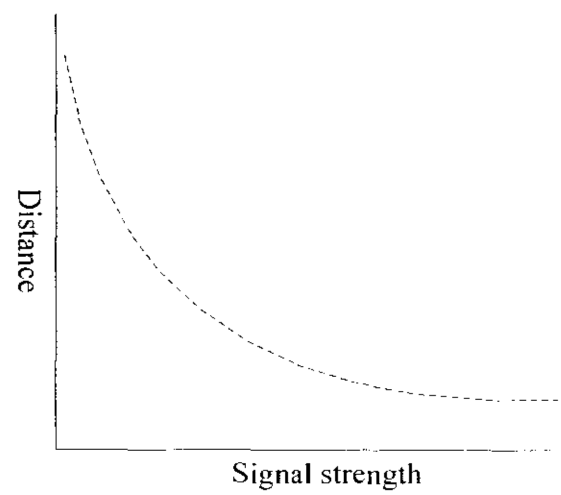


Figure 3.4: Distance V.s. the received signal strength

3.2.1.4 Weak Routing Link

A routing link between two directly connected devices is considered as a weak link if the devices have weak signal strength or they are out of the radio range of each other and cannot communicate properly. Fig.3.5 (a) shows the normal link while Fig.3.5 (b) shows the weak link between a master and a slave device and Fig.3.5 (c) shows two weak links between the source and the destination devices because of having a weak bridge device.

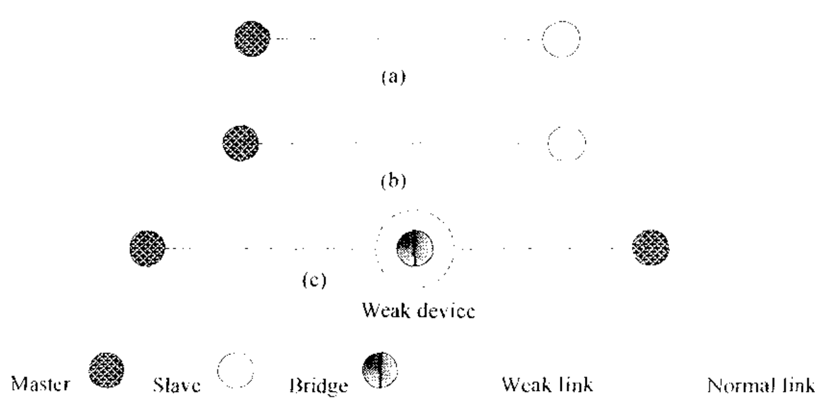


Figure 3.5(a): Example of the normal link between the source and destination, (b): Example of a weak link between the source and destination and (c): Example of a weak device between two weak links

3.2.1.5 Weak Routing Device

UCBT (F.J. Gonzalez-Castano and J. Garcia-Reinoso 2003) presents four levels of energy, i.e., L_0, L_1, L_2 and L_3. The proposed ERMP defines a threshold value for the energy level, if the energy level of any device gets to L_t and the master device predicts that the energy level of the device has decreased to the critical level (L_t), it selects a FBD from the table and makes a new link. In the routing path, a device which has the minimum energy level and weak signal strength is considered as a weak routing device, or a device whose energy level is normal but has weak signal strength is also considered as a weak device. It can be a sender or receiver.

3.2.1.6 Device Information Table (DIT)

According to the ERMP, within a scatternet, when a new link is established then the routing master and bridge devices maintain their information tables. The routing master device maintains a Routing Master Information Table (RMIT) that contains the list of slave devices, clock offset, device ID, device status (active/inactive), energy level, signal strength, Fall Back Masters (FBM) and Fall Back Bridges (FBB). Each routing bridge device maintains a Routing Bridge Information Table (RBIT) that contains the list of the connected master devices, device status (active/inactive), signal strength, energy level, list of FBBs, list of FBMs and the degree.

3.2.2 Route Maintenance Protocol

The proposed protocol ERMP is implemented in the dynamic Bluetooth network that forecasts the mobility of the devices because it needs to improve the network stability. In a dynamic network, devices can move any time so the Random Walk Mobility Model (RWMM) has been used for the network of the proposed protocol because it considers the speed and direction of the devices. When a device starts moving then ERMP monitors the device mobility through the RWMM. The RWMM is a simple mobility model based on the random directions and speed from a predefined range. In this model, any mobile device can move from its present position to a new position by randomly selecting the directions and speed. Suppose for each attached n = 1, 2, 3..., an area is divided into n × n sized units. From Fig.3.6, each n^2 area is known as a unit which is recognized by a pair.

$$(x, y) \quad x, y \in \{0, 1, 2 \ldots n-1\} \tag{3.10}$$

The time is divided into time slots t = 0, 1, 2... where at first, the t = 0 device is considered to be in any n^2 (x, y) unit where it can choose one of four adjacent units $\{(x+1, y), (x-1, y), (x, y+1), (x, y-1)\}$ with the same probability of 1/4 independently of the previous, and moves to the chosen unit at time slot $t = 1$. The node then replicates this procedure in every subsequent time period. The position of a device at timeslot $t = 0, 1, 2\ldots$ is denoted by $C_{(n)}(t)$, which basically shows the unit where the device exists (Emre and Oznur 2006).

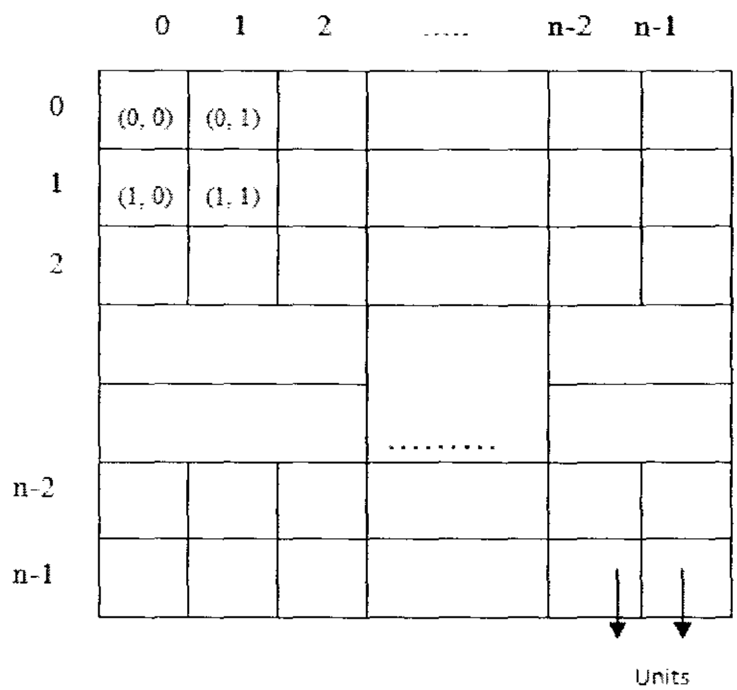


Figure 3.6: The Random Walk Mobility Model

The proposed protocol has two constraints of prediction:

3.2.2.1 Mobility prediction

The proposed ERMP starts by finding a communicating pair of source and destination. It makes a route between source and destination called the main route. Once the routing links have maintained, then the devices can communicate through their main routes. In the next phase, each routing master and routing bridge device maintains their tables. During transmission, a link breakage is predicted if any intermediate device between any pair of source and destination starts moving or fails. When an intermediate device starts moving from a piconet, then at the same time a routing master device can predict by its weak signal strength that the device is going out of its radio range.

The equation used for calculating signal strength is explained below.

$$PRe = PTr * GRe * GTr * (1 / (4 * \pi * d))^2 \tag{3.11}$$

where

GTr = *Transmitter antenna gain*

GRe = *Receiver antenna gain*

PTr = *Transmit power*

d = *distance*

where

$$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \tag{3.12}$$

The proposed protocol has been implemented in NS-2 (Network Simulator) because this simulator executes many propagation models to forecast the signal strength between the devices. The signal strength was used to monitor the successful frame transmission. The proposed protocol used the threshold value for the prediction of the signal strength as the simulator also had the ability to set a threshold value for monitoring whether the transmitted frame was received correctly.

According to the proposed protocol, all of the routing master devices and routing bridge devices keep the information of other connected devices in their tables. Each routing master device maintains its table in which it must keep the information of the signal strength (from the master device to the connected slave device), device status, energy level, clock off-set and the role of the device (Fall Back master, Fall Back Bridge). The routing bridge device keeps the information of the device ID, device status, signal strength (from the bridge device to the connected master devices), energy level and the role of the device (Fall Back master, Fall Back Bridge) in its table. As a threshold value is defined for the signal strength, if any intermediate device between the routing link starts moving, then the signal strength between the routing master and moving device is gradually decrease so the routing master device can easily predict, by its weak signal strength, that the device is moving and it can choose an appropriate routing device to make a new link before the main route breaks. This protocol also has the ability to change the role of the devices. For example, a routing bridge device is moving so the routing master device checks for a FBB device; if it is available, then the routing master device chooses it and maintains the routing link otherwise the routing master device chooses another device (that may be a slave device) and changes the role of the device as required. The new device must be within the radio range of the next and previous device so it can maintain the routing link efficiently.

Any routing device slave, master or bridge can be weak. If a device becomes weak and at the same time there are slave, master and bridge devices available as the FBDs, then the ERMP gives the highest priority to the bridge device because it can connect multiple devices from the scatternet. The second priority is given to the master device because it can connect multiple slave devices from its piconet. The third priority is given to the slave device if there is only a slave device available.

3.2.2.2. Analytical analysis of ERMP

This protocol has the ability to change the role of the devices due to a specific requirement. The ERMP has been explained by the example in Fig.3.7 (a); device A (source) and C (destination) are communicating through the route A-M_1-B_2-M_2-B_4-M_5-C and the other link is between device D (source) and E (destination) which are communicating through the route D-M_4-B_3-M_2-B_6-M_3-E. During transmission between A and C, the bridge device B_2 starts moving upward; in the meanwhile, the routing masters, M_1 and M_2, predict that the bridge device B_2, is moving so they search for the FBD as they already have the list of FBDs in their tables. Both of the routing master devices find device B_7 which is also a subset of their piconets. Both master devices transmit a request packet to device B_7 to make a new link between M_1 and M_2. This protocol has to make a new link before the main route breaks. As in Fig.3.7 (b), the new link between M_1 and M_2 is established and the data will go through the new route.

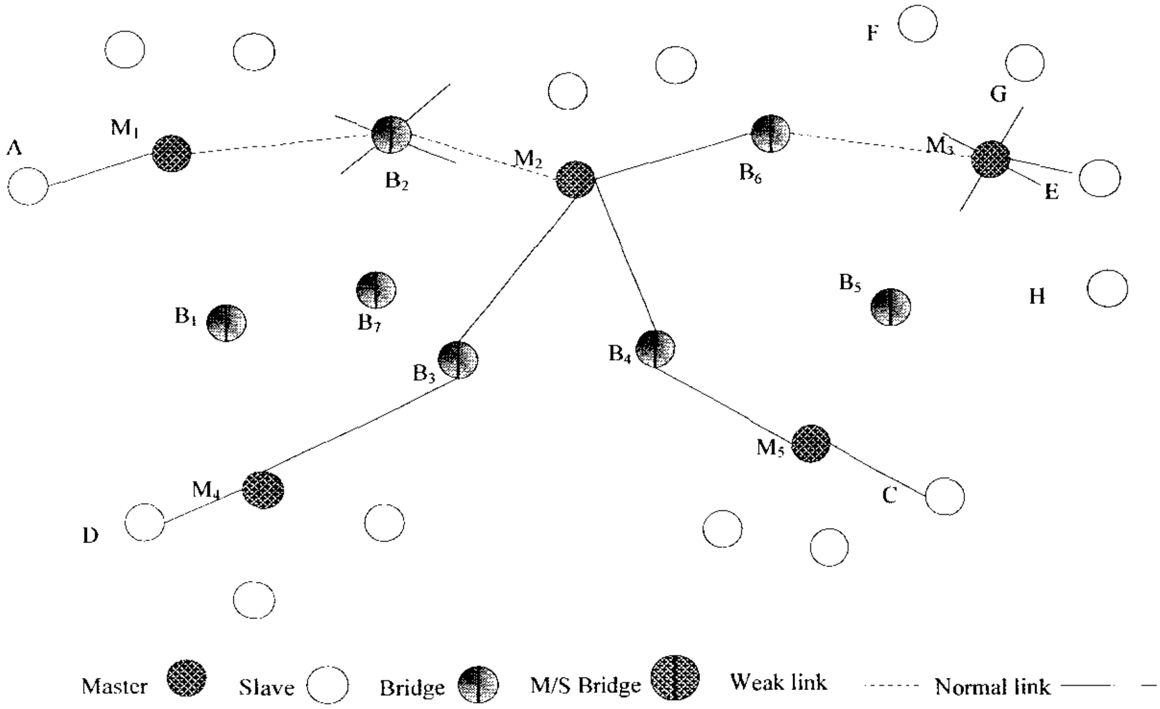


Figure 3.7(a): An example of predicting weak links

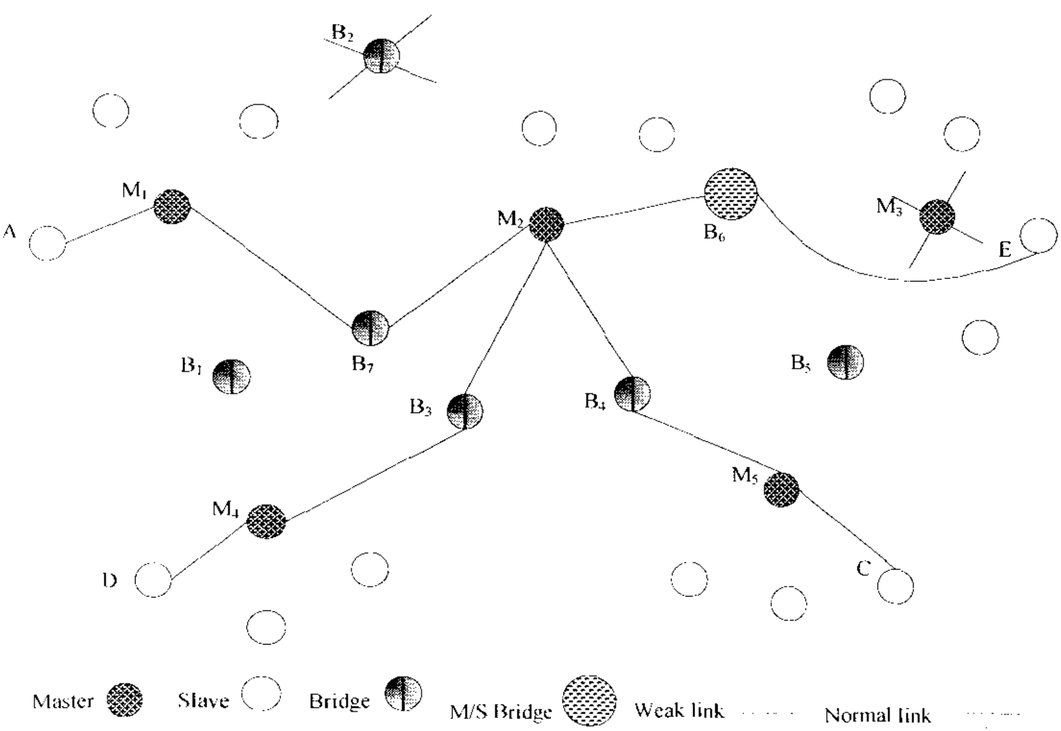


Figure 3.7(b): Route recovery by using the ERMP

Table 3.6: Routing Master Information Table (RMIT) for M_3

Device ID	Clock off-set	Device status (active/inactive)	Signal strength (dBm)	Energy level	Device role
E	c-offset(E)	Active	$S^t_{M3-E} = -70$	L_3	Slave
F	c-offset(F)	Inactive	$S^t_{M3-F} = -65$	L_2	Slave
G	c-offset(G)	Inactive	$S^t_{M3-G} = -76$	L_2	Slave
H	c-offset(H)	Inactive	$S^t_{M3-H} = -66$	L_3	Slave
B_5	c-offset(B_5)	Inactive	$S^t_{M3-B5} = -40$	L_1	Bridge
B_6	c-offset(B_6)	Active	$S^t_{M3-B6} = -40$	L_3	FBM (M/S bridge)

Table 3.7: Routing Bridge Information Table (RBIT) for B_2

Device ID	Device status (active/In active)	Signal strength (dBm)	Energy Level	Device role
M_1	Active	$S^t_{M1-B2} = -45$	L_2	master
M_2	Active	$S^t_{M2-B2} = -50$	L_2	master

On the other side of the same Fig.3.7 (b), when device D and device E are communicating, the Energy level of M_3 is gradually decreasing. Device M_3 checks for the destination device E as whether it is within the range of bridge device B_6 or not. Since device E is within the range of B_6, M_3 sends a request packet for the role switch operation to B_6. As in Fig.3.7 (b), B_6 switched its state from bridge to master/slave bridge. Now, B_6 has made a new link between B_6 and E. Through the prediction of the weak routing link and weak routing device, the ERMP does not require more time for re-establishing the link. Table 3.6 and Table 3.7 are the examples of the RMIT and the RBIT that give an idea about how the routing master and routing bridge devices maintain their tables. From Fig.3.7 (a), M_3 is the routing master device and B_2 is the routing bridge device. Whenever a weak link or weak device is predicted, then their corresponding routing master and bridge devices must update their tables.

3.2.2.3. Energy level prediction

The other constraint of predicting link breakage is the energy level which shows the fraction of the remaining energy (Roger M et al. 2005). The proposed ERMP is also an energy efficient protocol. During the main route construction, the master device must attach its available battery level to the route request packet and forward it to the connected bridge devices. The bridge devices also attach their battery level to the route request packet and forward it to the connected master devices. This procedure carries on until the route request packet arrives at the final destination.

The ERMP considers the general formula of the Master's average Power, MP_{avg}, given below:

$$MP_{avg} = \frac{i_{Tx} \times 1/2 T_{sniff} + i_{Rx} \times 1/2\, T_{sniff}}{T_{sniff}} \times V \quad (3.13)$$

$$MP_{avg} = ½\, (i_{Tx} + i_{Rx}) \times V \quad (3.14)$$

where Tx and Rx are the transmission and receiving slots. The voltage of the Bluetooth specific chip is indicated by V, and T_{sniff} is the sniff mode interval.

The slave device's Rx uses the average power SP_{avg}:

$$SP_{Rx} = \frac{(N_{sniff\_attempt} \times t_{slot}) \times i_{Rx}}{T_{sniff}} \times V \quad (3.15)$$

The slave device's Tx uses the average power SP_{avg}:

$$SP_{Tx} = \frac{i_{Tx} \times t_{slot} + ((N_{sniff\_attempt} - 1) \times t_{slot}) \times i_{Tx\_idle}}{T_{sniff}} \times V \quad (3.16)$$

where i_{Tx} is a task of slave device when it transmits and i_{Tx_idle} is a task of a slave device when in idle mode or not transmitting.

Each routing master and bridge device updates its table after the main route construction. If the energy level of any intermediate routing device is gradually decreasing and reaches the critical limit, then the routing master device updates its status as a weak device and chooses another appropriate FBD to maintain the routing link between the source and destination. If data is coming from the bridge device to the weak master device and the master device has to forward this data to the slave or bridge device, then a weak master device will check if the next device is within the radio range of the bridge device or not. If the next device is within the radio range of the bridge device, then the weak master device sends a request message to the bridge device to switch the role from bridge to master/slave bridge and make a direct link to the next device before the weak master device dies.

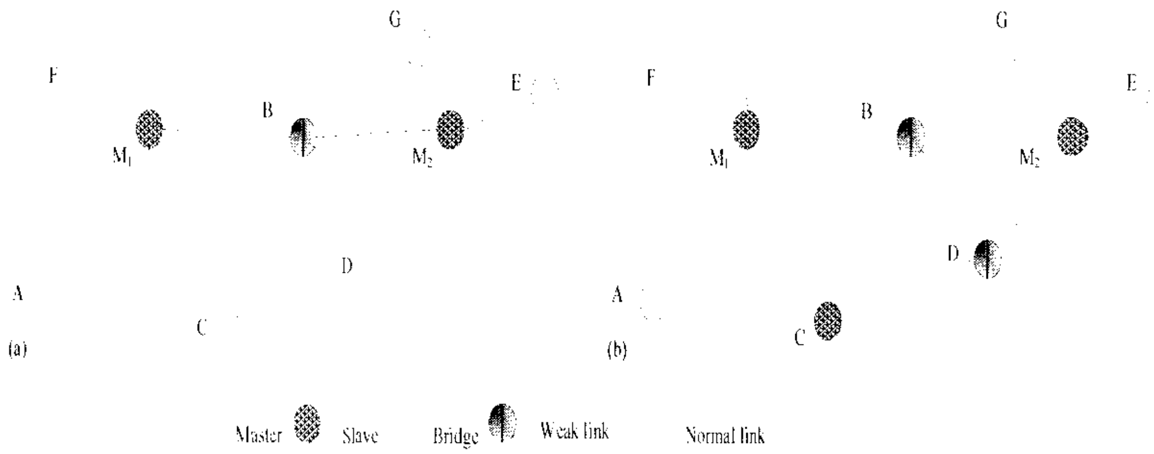


Figure 3.8(a): Prediction of weak link and weak device (b): Role switch operation for route maintenance

As shown in Fig.3.8 (a), devices, M_1 and M_2 predict that device B starts moving from its position and it is also predicted that the energy level of M_1 is gradually decreasing so that the routing link between $M_1$and M_2 becomes weak. In this case, when a master device itself becomes weak, it first of all finds a device that will become the master device; so, from its table it selects device C and sends a request for the role switch operation from slave to master. Now, the device C becomes a new master device so it updates its table. The master device C finds device D within its radio range and also within the range of M_2 so, device D works as a bridge between these piconets. For route maintenance, devices C and D make new links. In Fig.3.8 (b), the new routing link is A-C-D-M_2-E and data is going through the new route.

This protocol recovers the weak links efficiently and at the same time it achieves various advantages: increasing scatternet reliability, increasing packet delivery rate, reducing the number of control messages, reducing the time required for link repairing and improving throughput.

Algorithm 4: Route Maintenance Protocol

Notation:	**Procedure:**
Src: Source node	Connect( ): Activate new link
Dst: Destination node	Transmission( ): Activate communication
N: Total number of nodes	Req_role_swch ( ): Change device role
n: node	
B: Bridge	
M: Master	
RBIT: Routing Bridge Information Table	
RMIT: Routing Master Information Table	
RSP: Route Search Packet	
RRP: Route Reply Packet	
RR: Route request	
SNR: Signal-to-Noise Ratio	
EL: Energy Level	

Input: *Routing Master and Routing Bridge Information Table*
Output: *Activate the new route*

1. ***foreach*** $n_i \in N$ ***do***
    1.1. *Network Connectivity (Define Master and Slaves)*
    end foreach
2. *Src Initiate RR and forward RSP to Master*
3. ***do***
    3.1. ***foreach*** $n_i \in B$ ***do***
        3.1.1. $RSP_{(ID,\ Role)}$
        end foreach
    3.2. ***foreach*** $n_i \in M$ ***do***
        3.2.1. $RSP_{(ID,\ Role)}$
        end foreach
    while *(!=Dst)*
4. *Send RRP to Source*
5. *Routing Master and Routing bridge update RMIT and RBIT, respectively*
6. ***do***
    6.1. ***if*** *SNR* $< \Theta$ ***or*** $EL=L_1$, ***then*** where $\Theta = -45$db
        6.1.1. *Select FBD*
        6.1.2. ***if*** *(Send activation request( )=Null)* ***then***
            6.1.2.1. *Req_role_swch( )*
            endif
7. *Update (RMIT)*
8. *Connect( )*
    end if
9. ***while*** *(transmission( ))*

In the first step, a fully connected Bluetooth network is established. In the second step, routing links are defined for the pairs of source and destination nodes, where the source node initiates the route request for the destination node. The source node forwards a RSP to the master node and the master node forwards the received packet to the connected bridges in the third step, until the RSP reaches the destination node. Once the destination node receives the RSP it sends the unicast RRP to the source node in step 4. Hence, the transmission starts, if the routing master receives any packet that the SNR is less than Θ, in step 6, or the energy level of the device reaches L_1. The master declares the link as a weak link and activates the FBD in line. If there is no FBD, the master performs the role switch operation and establishes a new link in step 8. Finally, the routing master activates a new link and forwards data through the new link.

3.3. AN EFFICIENT COMBINED CONGESTION HANDLING AND ROUTE MAINTENANCE PROTOCOL FOR DYNAMIC ENVIRONMENT IN BLUETOOTH NETWORK (CHRM)

In this section, proposed congestion handling and route maintenance protocols were merged into a new combined protocol CHRM. Many scatternet optimization protocols have been implemented but during transmission sometimes they created some serious problems of congestion or route maintenance. These problems ultimately degraded the overall network performance. The main problem with previous protocols is that once the main link broke then they presented the solution for route maintenance that apparently consumed more time, more energy and more control packets. Fig.3.9 clarifies these problems in detail.

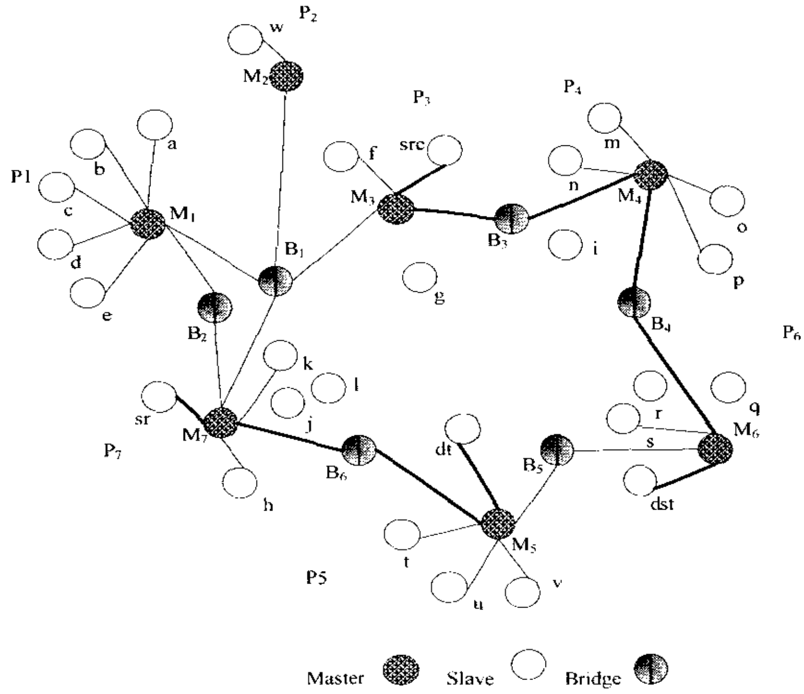


Figure 3.9: Example of scatternet congestion and route maintenance problems

Suppose piconets, P_1 and P_5, are considered as congested because the devices inside the piconets are communicating more frequently and ultimately creating the intra-piconet congestion. Piconets, P_1, P_2, P_3, and P_7, are communicating through a single bridge device B_1. B_1 is used more frequently so it has congestion known as inter-piconet congestion. Devices having congestion decrease the network life time and cannot communicate efficiently.

In the same Fig.3.9, two routes (src-M_3-B_3-M_4-B_4-M_6-dst) and (sr-M_7-B_6-M_5-dt) have problems of route maintenance. During transmission, the bridge device B_3, starts moving and the link between M_3 and M_4 becomes weak. On the other side, the energy level of M_7 starts decreasing and reaches the critical level. In both of these cases, the mobility and minimum energy level of the devices can break the routing link and create major routing drawbacks. Many techniques have been proposed for solving the problem of congestion and route maintenance but they still have to be improved.

In this thesis, to overcome the problem of intra-piconet and inter-piconet congestion the ECHP protocol has been proposed and the ERMP protocol has been implemented to solve the problem of route breakage. After initializing these protocols, they maintain tables for the master and bridge devices. So, the CHRM also maintains the tables for the master and bridge devices. This is a first protocol that provides the solution for congestion handling and route maintenance in the same protocol. This protocol was implemented because during transmission some time when congestion is handled then problem of route maintenance occurs and when sometimes route maintenance is handled then the problem of congestion occurs. Therefore we implemented a new protocol that considers the both problems of congestion and route maintenance at the same time and provide solution.

3.3.1. Analytical analysis of CHRM

From Fig.3.10, as the devices in P_1 and P_5 are communicating more frequently with each other, the proposed CHRM shares the intra-piconet traffic load by using two techniques PFP and SFP; as well, it also shares the inter-piconet traffic load on device B_1 with the help of the FBB. The CHRM protocol predicts the weak routing links and the weak routing devices on the basis of mobility and energy level. When it predicts a weak link or weak device it activates the FBD before the main route breakage. Table3.8 is the example of master information table and Table3.9 is the example of bridge information table.

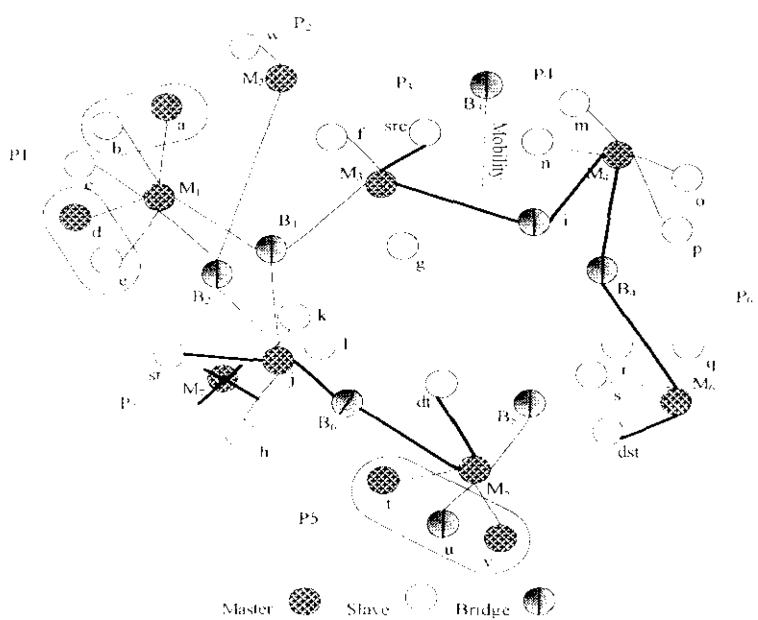


Figure 3.10: handling of congestion and route maintenance by using CHRM

Table.3.8. Master Information Table for M_7

Device ID	Clock off-set	Device Status	Signal strength	Energy level	Device role	Transmit data	Received data
sr	C-offset(sr)	active	$S^t_{M7\text{-}sr}$= -70	L_2	Slave	300	200
h	C-offset(h)	active	$S^t_{M7\text{-}h}$= -67	L_3	Slave	200	100
k	C-offset(k)	active	$S^t_{M7\text{-}k}$= -66	L_2	Slave	200	289
j	C-offset(j)	active	$S^t_{M7\text{-}j}$= -80	L_3	FBM	200	50
B_1	C-offset(B_1)	active	$S^t_{M7\text{-}B1}$= - 85	L_2	Bridge	100	400
B_2	C-offset(B_2)	active	$S^t_{M7\text{-}B2}$= -75	L_3	FBB	50	156
B_6	C-offset(B_6)	active	$S^t_{M7\text{-}B3}$= -85	L_3	Bridge	200	40

Table.3.9. Bridge Information Table for B_1

Device ID	Device status	Signal strength(dBm)	Energy level	Device role	Transmit data	Received Data
M_1	active	$S^t_{M1\text{-}B1}$= -80	L_2	Master	300	270
M_2	active	$S^t_{M2\text{-}B1}$= -70	L_3	Master	80	90
M_3	active	$S^t_{M3\text{-}B1}$= -69	L_3	Master	40	50
M_7	inactive	$S^t_{M7\text{-}B1}$= -78	L_1	Master	500	497
j	active	$S^t_{j\text{-}B1}$= - 85	L_3	FBM	400	200

The CHRM performed the role switch operation for the most frequently communicating pairs. For example, (b, a), (d, e) and (t, v) are communicating, repeatedly. The proposed CHRM used the slave devices (a, d, t, and v) as auxiliary masters and the device u as the auxiliary bridge to share the congestion on M_1 and M_5. In piconet P_1 it performed the PFP and in piconet P_5 it performed the SFP to overcome the congestion problem on the master devices. The bridge device B_2, had congestion and it did not allow the parallel transmissions between the piconets so the proposed protocol has started the transmission by using the FBB, B_2. By activating the FBB, it allowed the parallel transmissions between the piconets and shared the congestion on a single bridge device. In case there was no alternate bridge device, then this protocol could perform the role switch operation to make a new bridge. During transmission, the CHRM monitored the mobility and energy level of the devices so when it predicted the critical level it activated alternate devices to prevent the route breakage. For example, in the same Fig.3.10, during transmission between source device (src), and destination (dst), the intermediate routing bridge device start moving from its position; when it reached the critical level, the master devices, M_3 and M_4, predicted their mobility and the weak link from their tables and at the same time, they activated the device, i, as a bridge and made a new link between M_3 and M_4. On the other side, during transmission between the source (sr), and destination (dt), the energy level of the routing master, M_7 in P_7, was gradually decreasing which made it as a weak device; when it predicted its energy level is decreasing, at the same time it activated the FBM device and started transmission through the FBM device. The main purpose of the CHRM is to prevent the route breakage and activate a new link before the main route breakage; therefore, it predicts the device energy level and mobility.

The combined CHRM protocol was implemented and compared with the previous protocols (DCC and ROMA). After getting the simulation results, it was observed that the CHRM protocol outperformed the previous DCC and ROMA protocols.

3.3.2. Steps of CHRM protocol

1. Firstly, the master node enters into the inquiry state and the slave nodes enter into the inquiry scan state.
2. Secondly, the master node executes the paging and the slave nodes execute the page scan operations for the synchronisation.
3. In the third step, a constructed and connected scatternet is considered.
4. In the forth step, the master and bridge devices maintain their tables.
5. In the fifth step, the threshold value for the traffic load, energy level and signal strength are defined. As well, the routing links are established.
6. In the sixth step, monitoring of the MF communicating pairs, congestion, energy level and mobility of the devices are performed.

   If congestion is predicted on a master device, then it performs PFP or SFP:

a) The master node finds the MF communicating pair of the source and destination.

If they are directly within the radio range of each other then CHRM performs PFP.

b) The master node sends a request packet for the role switch operation to the pair of the source and destination.

c) The master node updates the record of the source and destination as in the park mode.

d) The source node is selected as an auxiliary master.

e) An efficient PFP is created for direct communication between devices that shares the congestion on a master node.

7. If the devices are not within the radio range of each other then CHRM performs SFP.

   a. The master node checks an intermediate node between the source and destination.

   b. The master node sends a request packet of the role switch operation to the intermediate node and to the pair of the source and destination.

   c. The master node selects an intermediate node as AxB in the same piconet and both the source and the destination nodes are selected as AxMs.

   d. The master node updates the record of the AxB and the pair of the AxMs as in the park mode.

   e. Both the AxMs make connection with the AxB and start efficient communication without increasing delay.

8. If congestion is predicted on a bridge node, then it activates a FBB.

   a) Firstly, the bridge node sends a request packet to the connected masters for the FBB.

   b) The master nodes reply with the information of the FBB that can be used for congestion handling.

   c) In this step, master nodes update their status and transmit a request packet of activation to the FBB. Now, the FBB enters into the park mode. The FBB makes temporary links between the piconets.

d) Finally, the piconets can communicate through the FBB and efficiently shares the congestion.

9. After successful communication, the temporary links will be disconnected and the devices will return to their original states for further use.
10. During transmission, if a device having a critical energy level or a device mobility is predicted, and then the CHRM replaces the weak device or weak link. During transmission, the mobility of the devices can be predicted by their weak signal strength and the energy level through critical energy level. If weak links or weak devices are predicted, then:
    a. The routing master device selects a FBD that already exists in its table.
    b. The routing master device transmits a request packet to the FBD for the new route activation.
    c. The FBD has to make a new link before main route breakage. If the FBB is not available, then the CHRM performs a role switch operation.
    d. The master device disables the weak links between the devices and updates the table.
    e. Finally, data start going through the new route.
11. In case a master device starts moving, then, first of all, it will inform all of the connected slave devices and activate its FBM. On the other hand, if the energy level of the master device starts decreasing to L_1, then the master device sends a request message to the FBM device to become a new master and establish a new link before the weak link breaks.

The Fig.3.11 shows the system model of CHRM.

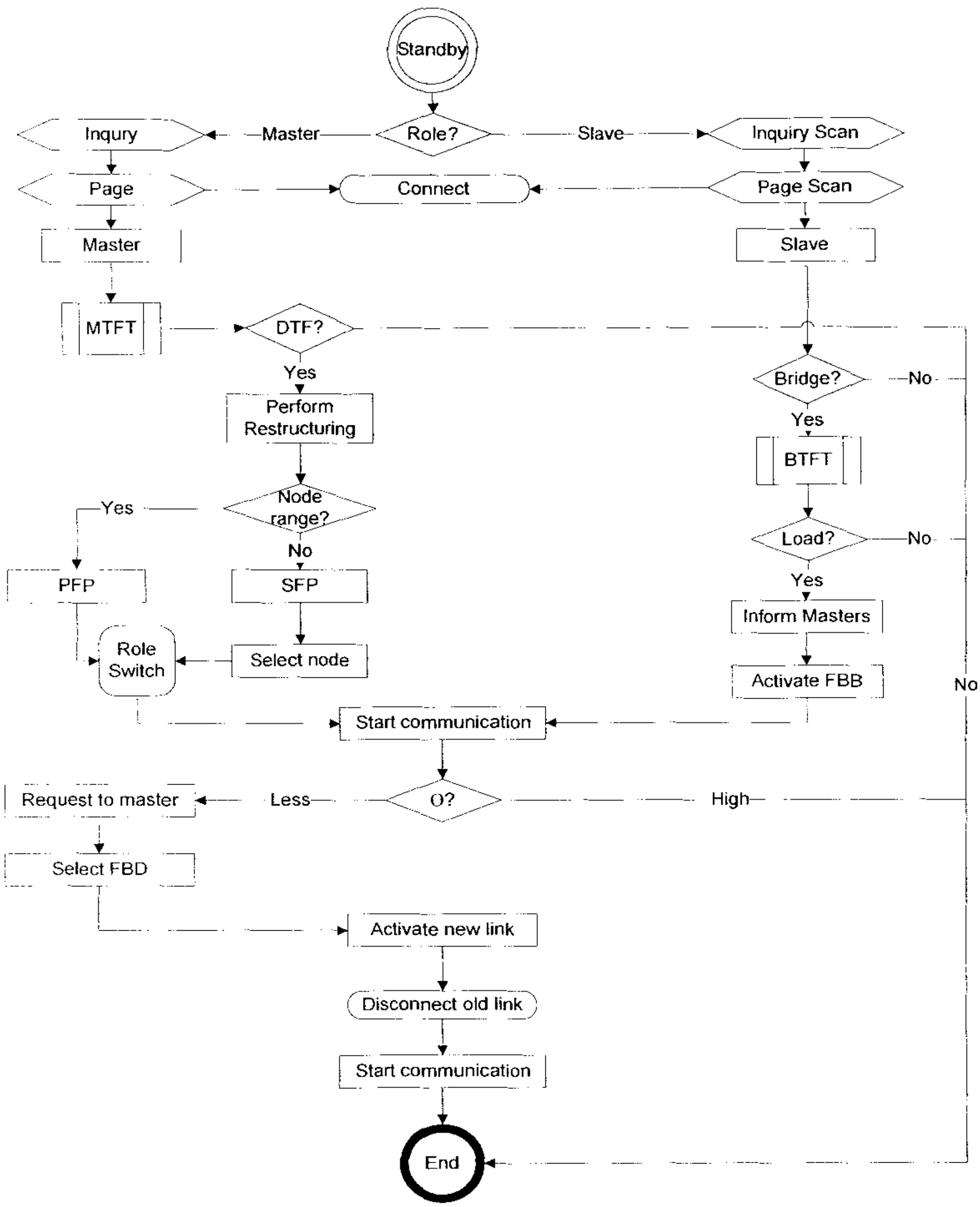


Figure 3.11: System model for CHRM

3.4. SUMMARY

In this chapter the proposed protocols have been explained. The first protocol is An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP) that shows how the congestion can be handled. This is because of an inefficient Bluetooth topology may create a bottleneck and delay in the network when data is routed. The PRP frequently performs network restructuring which creates more control overhead. In contrary, the DCC protocol does not consider real network restructuring it depends on a backup node; it has been observed that in many cases the backup node is hard to find. Moreover, the DCC protocol uses the centralized approach; therefore, it does not truly identify the congestion on the bridge node. The proposed ECHP protocol shares the traffic load on the master node by network restructuring and shares the traffic load on the bridge node by creating a FBB. During the piconet restructuring, the ECHP protocol performs the PFP and SFP. The PFP reconstructs a new piconet in the same piconet for the source and destination which are directly within the radio range of each other. On the other hand, the SFP reconstructs the scatternet within the same piconet if the source and destination nodes are not within the radio range of each other. The ECHP protocol creates a FBB within the scatternet; this reduces the data congestion on bridge node and provides an efficient way for parallel transmission. Analytically, the ECHP has efficiently shared the network traffic load and increased the network throughput.

The second proposed protocol is the Efficient Route Maintenance Protocol for dynamic Bluetooth networks (ERMP). Initially, an active and connected Bluetooth scatternet was considered and maintained the routing links. After the main route construction, each routing master and routing bridge device kept the information of all the connected devices. The proposed protocol predicted the weak links and weak devices through the signal strength and energy level of the devices and repaired the weak links by activating the FBDs. It was analyzed that the ERMP was more efficient than ROMA and DENM because both of the protocols provided a solution for route breakage only when the main route has been broken. On the contrary, the proposed protocol predicted the weak routing links and weak devices and if a weak routing link was predicted, then this protocol made a new link before the main route broke. The simulation results revealed that the ERMP outperformed in terms of average healing delay, control packet overhead, energy consumption, data packet loss, number of successful repaired links and throughput.

At the end, these two proposed protocols were combined into one new protocol, "An Efficient Combined Congestion Handling and Route Maintenance Protocol for

Dynamic Environment in Bluetooth Network" (CHRM), which considers the existing scatternet congestion and route maintenance problems and their solutions in a same protocol. The CHRM protocol was simulated and the results were compared with the previous protocols.

CHAPTER 4

RESULTS AND ANALYSIS

This chapter covers the detailed evaluation of the methodology. The proposed ECHP, ERMP and CHRM protocols have been implemented using simulation; therefore, the simulator and its parameters are described at first then the comparison results of the proposed and previous protocols are discussed. The result part contains further three parts: the first part contains the results of the ECHP protocol which has been compared with the previous PRP and DCC protocols, the second part contains the results of the ERMP which has been compared with the previous ROMA and DENM protocols and the third part contains the results of the CHRM protocol that is a combined protocol of the ECHP and ERMP protocols. The results of the CHRM have been compared with the results of the existing DCC and ROMA protocols.

4.1. SIMULATOR

This section briefly discusses the simulation architecture that was used for the evaluation of the previous and proposed protocols. Simulation is defined as "the method of designing a real system model and carrying out experimentation for the purpose of understanding and evaluating the performance of the system". A simulation is considered as a combination of science and art. The proficiency in mathematics and computer programming is considered for the science part, whereas the proficiency in the conceptual model formulation shows the art portion (Teerawat and Ekram 2009).

The proposed protocols, ECHP, ERMP and CHRM were implemented in NS-2 using the Bluetooth simulation model of the University of Cincinnati Bluetooth (UCBT) . The UCBT is an open source that provided an absolute simulation model for the Bluetooth performance evaluation. The UCBT has the built-in Bluetooth functions of inquiry, inquiry scan, page, page scan etc. It also supports the device mobility through the "setdest".

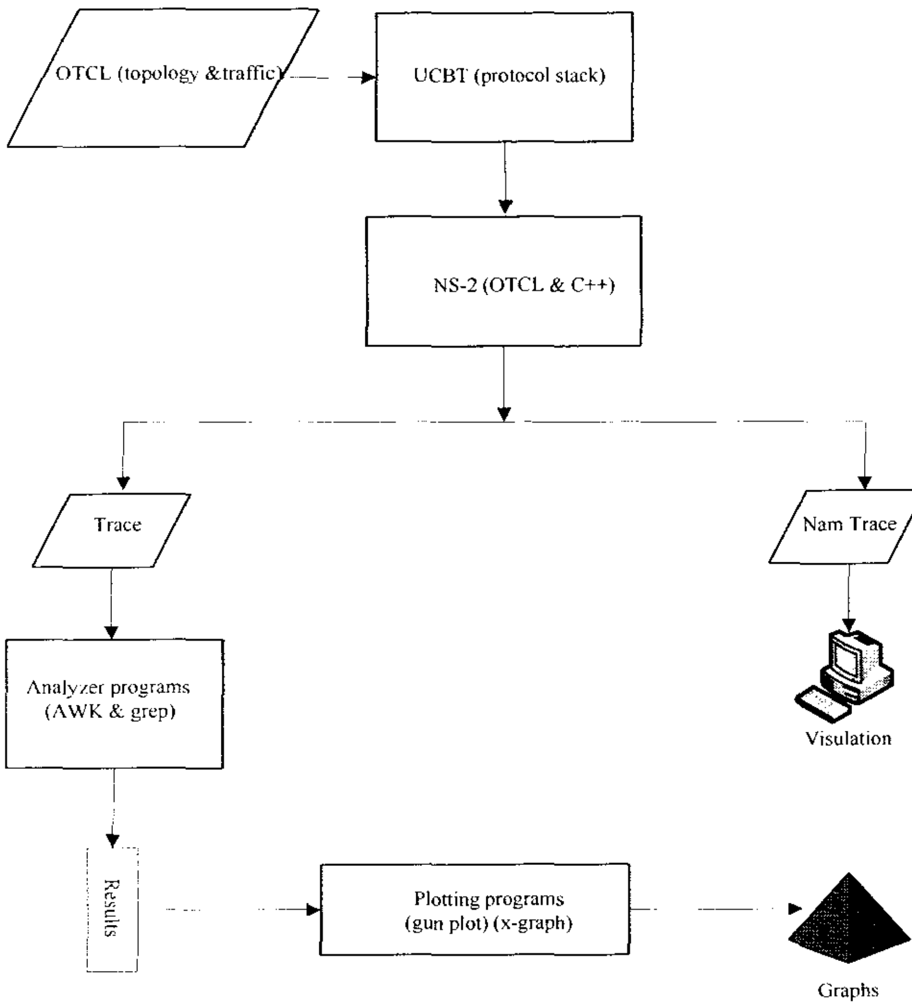


Figure 4.1: Simulation structural design

Fig.4.1 shows the simulation structural design. NS-2 generates the OTCL file for the simulation of a network because it defines the traffic and topology patterns. After a successful execution, it generates two output files Trace and Nam Trace. A Trace file generates analytical results while the animation is generated in a Nam Trace file. The AWK and grep commands are used for the desired data separation. After getting the results, graphs (gun plot or x-graph) can be plotted.

4.1.1. Simulation setup

The proposed and base protocols were implemented in the UCBT based on the NS-2 simulator. This section evaluates the simulation results by comparing numerous performance metrics: total delay, control packet overhead, energy consumption, data packet loss, successful recovered links, healing delay and throughput etc. We run the simulation ten times and then graphs were plotted.

In order to manage a simulation environment parameters were used as are listed in Table 4.1. The four-sided area of 80m x 80m was taken in which the Bluetooth devices were scattered randomly. During simulation, a total of 90 Bluetooth devices were used and 84 pairs of source and destination devices were selected randomly. The Random Walk Mobility Model (RWMM) was used as a mobility model. Data packet types DH3 and DH5 were used and the Round Robin, RR, algorithm was selected for scheduling. For performance estimation, the Constant Bit Rate (CBR) traffic model was taken to generate the data traffic for each routing link. The Maximum Distance Rendezvous Point was selected for bridge scheduling algorithm because it supports the mesh topology. The whole simulation run time was set for 1000 sec.

Table 4.1: Simulation parameters

Parameters	Values
Network dimension	80m x 80m
Number of devices	15 - 90
Traffic model	Constant Bit Rate (CBR)
Mobility direction	RWMM
Scheduling algorithm	Round Robin
Mobility speed	1 m/s
Packet size	358 kb
Packet interval	0.15 s
Queue length	50 packets
Number of device pairs	84
Data packet type	DH3, DH5
Communication range	10m
Energy consumption	0.0763×10^{-6} J/bit
Inquiry time	10.24s
Paging time	1.28s – 2.56s
Bridge scheduling algorithm	Maximum Distance Rendezvous Point
Simulation time	1000

4.2. RESULTS

In this sub-section, the simulation results are discussed in three parts; the first part contains the results of An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP) which was compared with the previous PRP and DCC protocols. The Second sub-section contains the results of An Efficient Route Maintenance Protocol which was compared to the simulation results of the previous ROMA and DENM Protocols. The third sub-section shows the results of An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in

Bluetooth Network. For getting desired result, the simulation was run ten times. After getting the comparison results, it was observed that the proposed protocols outperformed the previous protocols.

4.2.1. An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP)

An Efficient Congestion Handling Protocol for Bluetooth scatternet was designed for the congestion handling of a Bluetooth network (piconet, scatternet). To share the congestion on a master device, the PRP performed piconet restructuring, whereas the DCC protocol activated the backup device. When congestion arose on a bridge device, then DCC did not activate the backup device. The ECHP considered the congestion on a master device as well as on the bridge device, and it provided the solution for both the intra-piconet and inter-piconet congestion. To check the performance of the ECHP protocol, it was simulated and compared with the previous protocols. The protocols were compared in terms of total delay, energy consumption, packet loss; control overhead, Average number of links passing through a node and network throughput. The comparison results show that the ECHP protocol outperformed the previous PRP and DCC protocols. To evaluate the ECHP, DCC and PRP protocols, the performance is measured from performance parameters as listed in Table4.2. These parameters were selected because these parameters explain the efficiency of protocol. As ECHP protocol allow the parallel transmission between piconets so from parameter "average number of links passing through a node" shows the number of links passing through from each bridge.

Table 4.2: performance parameters

Parameters	ECHP	DCC	PRP
Total delay	√	√	√
Total energy consumption	√	√	√
Packet loss percentage	√	√	√
Control packet overhead	√	√	√
Average network throughput	√	√	√
Average number of links passing through a node	√	√	√

4.2.1.1.Total delay

The total delay stipulates how long it takes for a data bit to travel from a source device to a destination device. It is normally measured in fractions of seconds and may vary depending on the position of the exact pair of communicating devices. The total delay of the ECHP protocol was compared with the PRP and DCC protocols. More incoming Bluetooth devices in a scatternet prolong the polling time of the master nodes. According to the PRP, it performed piconet restructuring for all pairs of source and destination nodes. As it provided the solution for Congestion handling on the master node, therefore, it increased the network delay. On the other hand, the DCC protocol did not provide any solution when congestion occurred on a bridge node. When a single bridge node participated in multiple connections, it might have created the problem of congestion and delay. Whereas, the ECHP protocol shared the traffic load on the master node as well as the traffic load on the bridge node; this minimized the delay. Fig.4.2 shows the total delay of protocols and it is observed that the ECHP protocol had less delay as compared to the PRP and DCC protocols. The total delay can be calculated by using the following formula, where total numbers of received packets are *1* to *n*.

$$Total\ delay = \sum_{i=1}^{n}(\ packet\ received\ time - packet\ transmitted\ time) \quad (4.1)$$

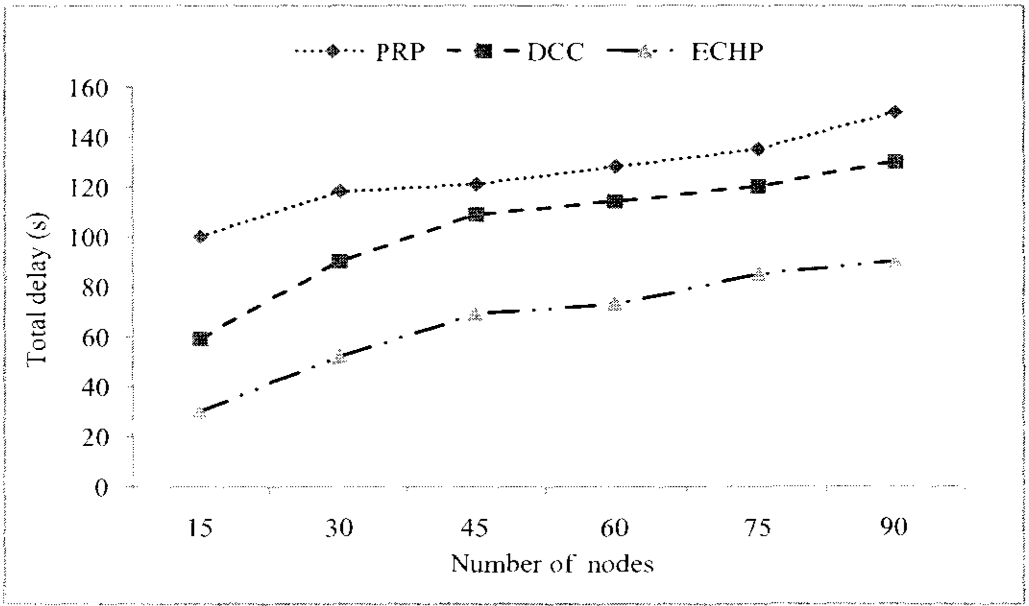


Figure 4.2: Total delay vs. number of nodes

4.2.1.2. Total energy consumption

The total energy consumption of the ECHP, PRP and DCC protocols was calculated and compared. As compared to the PRP and DCC protocols, the proposed ECHP protocol consumed less energy because the ECHP protocol efficiently shared the traffic load of the master and bridge devices. For intra-piconet Congestion handling, it performed network restructuring on those nodes which were communicating more frequently. Whereas, it handled the inter-piconet congestion through the FBB, which shared the traffic loads on the bridge node. As shown in Fig.4.3, it is observed that the DCC and PRP consumed more energy than the ECHP protocol because it avoided the extra guard time utilization; therefore, it consumed less energy. When PRP performed the network restructuring frequently, it consumed more guard time for synchronization. In the case of the DCC protocol, it did not allow the parallel transmissions between the piconets so the devices had to wait for the availability of the intermediate device. During wait time the devices consumed extra energy. Mostly, the Bluetooth devices consumed energy during the inquiry and the inquiry scan process. The energy consumption can be calculated by the following formula.

$$\textit{Energy consumption} = (\textit{number of received bits} + \textit{number of transmitted bits}) \times 0.0736 \times 10^{-6} \quad (4.2)$$

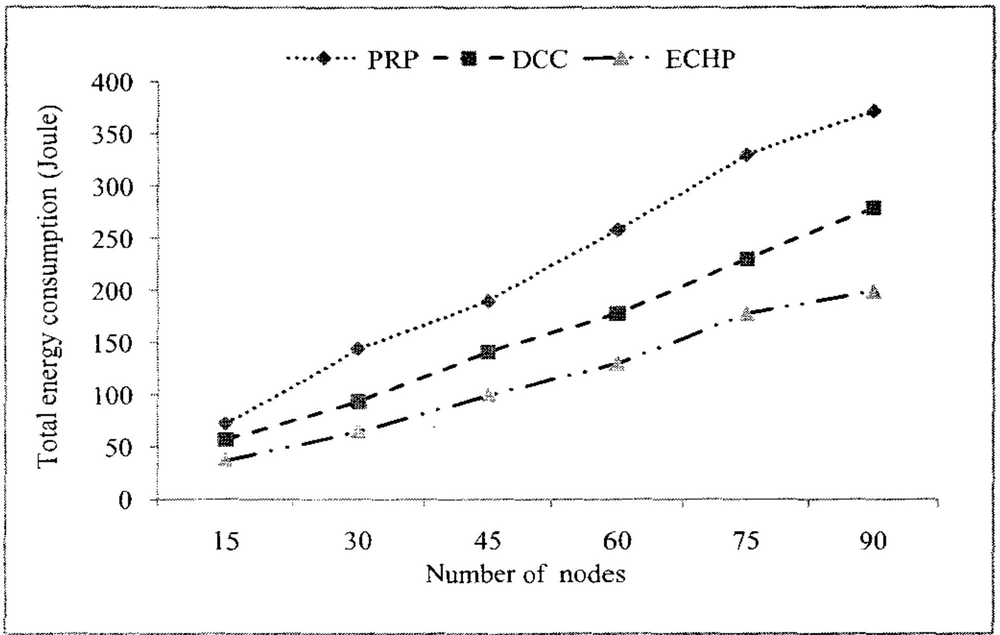


Figure.4.3: Total Energy consumption vs. number of nodes

4.2.1.3. Packet loss percentage

Packet loss happens when one or more data packets are travelling across a network and fail to reach their destination (Chung-Hsin and Yun-Mou 2009). The PRP made new piconets frequently within the piconet and if other devices wanted to communicate with the devices which had changed their roles, they could create the packet loss. Whereas, the DCC protocol did not support the congestion handling on the bridge node so during the communication chances were high for more packet loss. The ECHP protocol is an efficient protocol which created efficient links for intra-piconet and inter-piconet communication so it decreased the rate of the packet loss. When the ECHP protocol performed network restructuring, it changed the mode of the device into the park mode. After successful transmissions, it changed the mode of the devices into the original states. From Fig.4.4, it is perceived that the ECHP protocol performed better than the PRP and DCC protocol. The percentage of packet loss can be achieved by the following formula.

*Packet loss =Number of received packets /number of transmitted packets *100* (4.3)

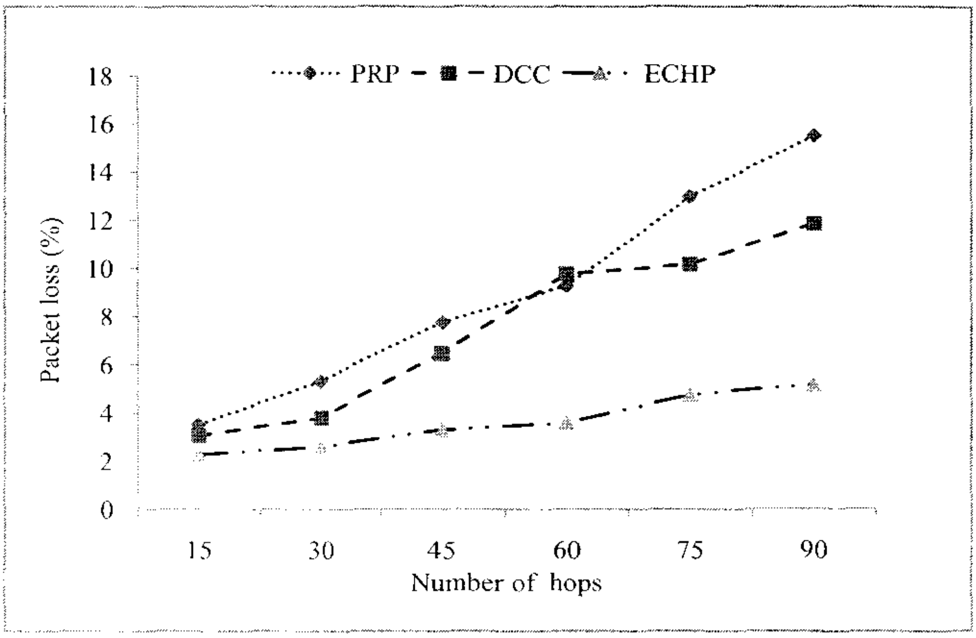


Figure 4.4: Packet loss percentage vs. number of nodes

4.2.1.4. *Control packet overhead*

The sum of the control packets used for the communication or the size of the control information is known as control overhead (Sheng-Wen et al. 2009). It can be computed as the sum of (MAC + bb + Null + POLL) packets for the received data packets. The control packet overhead of the PRP and DCC protocols was compared with the ECHP protocol. As the ECHP protocol utilized the resources efficiently and it did not perform the network restructuring frequently within the piconet, therefore, it used a lower number of control packets. As mentioned in Fig.3.1, suppose there are two piconets in a scatternet. During the communication, the masters (M_1, M_2) predict that devices (i and j) from P_1, devices (h and f) from P_1 and from P_2 devices (a and b) are communicating very frequently and creating intra-piconet congestion. Therefore, to reduce the congestion, the ECHP protocol performs the network restructuring by using the role switching operations. The PRP frequently creates new piconets within the piconet and makes new links so that each time during synchronization, the Bluetooth devices use extra control packets. Whereas, in the DCC protocol to overcome the problem of delay within the piconet it finds a backup device by utilizing extra control packets. Fig.4.5 shows that the PRP and DCC protocols utilize more control packets compared to the ECHP protocol.

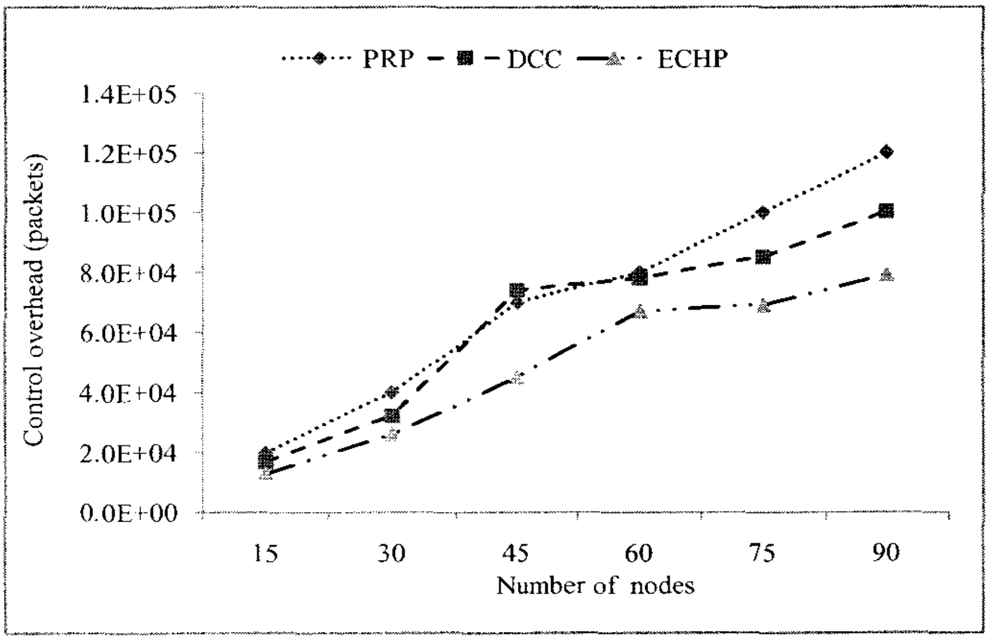


Figure 4.5: Control overhead vs. number of nodes

4.2.1.5. Average Network throughput

Network throughput is the rate of successful delivery of data packets per unit time over a network. Throughput is normally measured in bits per sec (bits/s) (Hassan et al. 2008; Pasolini and Verdone 2005). The throughput of the ECHP, PRP and DCC protocols was compared and it was observed from the comparison results that the ECHP protocol showed better results than the PRP and DCC protocols. The PRP and DCC protocols created delay, consumed more control packets and their packet loss rate was also higher than the ECHP protocol. It was observed that ECHP approach increased the number of data packets delivery and increased the network throughput. The ECHP allowed the parallel transmission because it could improve the network performance. The FBB was used when a single bridge node was not sufficient for an efficient communication between the piconets and when the parallel transmissions were required between piconets for a well organized and smooth communication. In Fig.3.3 (b), different piconets were communicating through $B_1$and due to the congestion, node (a) had been selected as the FBB for P_2 and P_3, and node (d) had been selected as the FBB for P_1 and P_2. From Fig.4.6 it has been predicted that the throughput of the ECHP protocol was higher than the previous PRP and DCC protocols but it decreased the throughput with the increase of the number of hops. The throughput can be calculated by the following formula:

Throughput = total number of received packets by the receiver device / total simulation time (4.4)

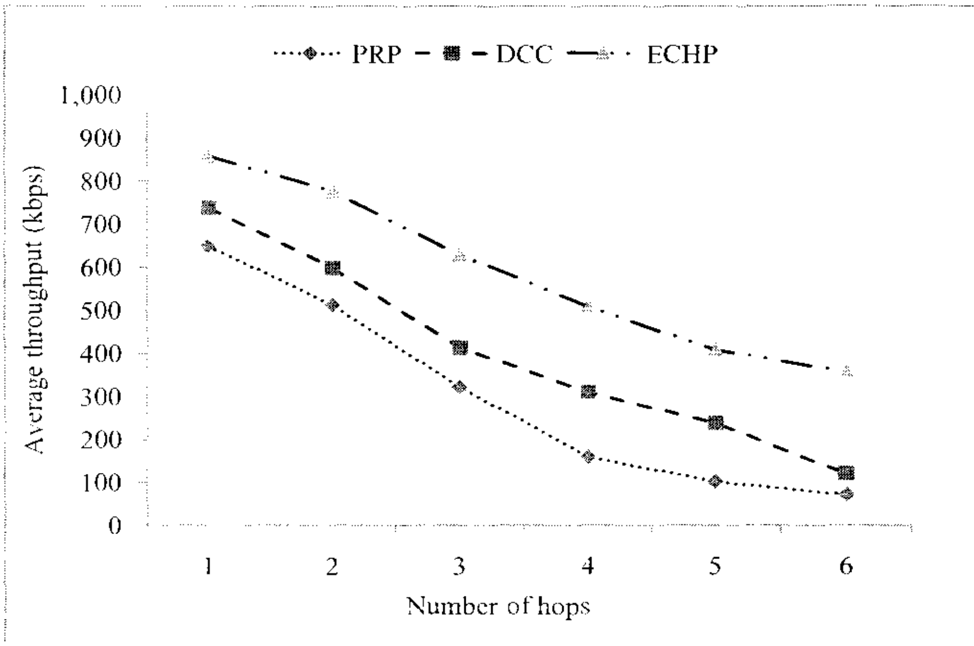


Figure 4.6: Throughput vs. number of connections

4.2.1.6. Average number of links passing through a bridge node

During the communication it was observed that, when the number of passing links increased through a single bridge node, the proposed ECHP protocol activated a Fall Back Bridge that shared the traffic load. It allowed the parallel transmission between the piconets that reduced the wait time of the other nodes. Whereas, when the same numbers of links are passed through a single bridge node by using the PRP and DCC protocols, they did not activate any device that could share the congestion on a bridge node. During this situation, when one pair of devices transmitted data at that time another pair could not transmit data. Devices of other pairs had to wait for the bridge node to become free. Since the DCC and PRP are both congestion handling protocols, the DCC has been proposed for the intra-piconet congestion handling. When congestion arose on a master node, then it activated a bridge node for load balancing, whereas the PRP solved the piconet congestion by creating more piconets within the piconet. The PRP created new piconets for all of the communicating pairs. From Fig.4.7, it is observed, that the ECHP shared the congestion more efficiently than the DCC and PRP. As the total number of available bridges is 18 and the total number of connections is 84, the PRP used fourteen bridges, the DCC fifteen and the proposed ECHP used seventeen bridge nodes. Hence, the total traffic load has been efficiently shared; this improved the overall network performance. The average number of links can be calculated by the following formula:

Average number of links = number of connections / number of bridges used (4.5)

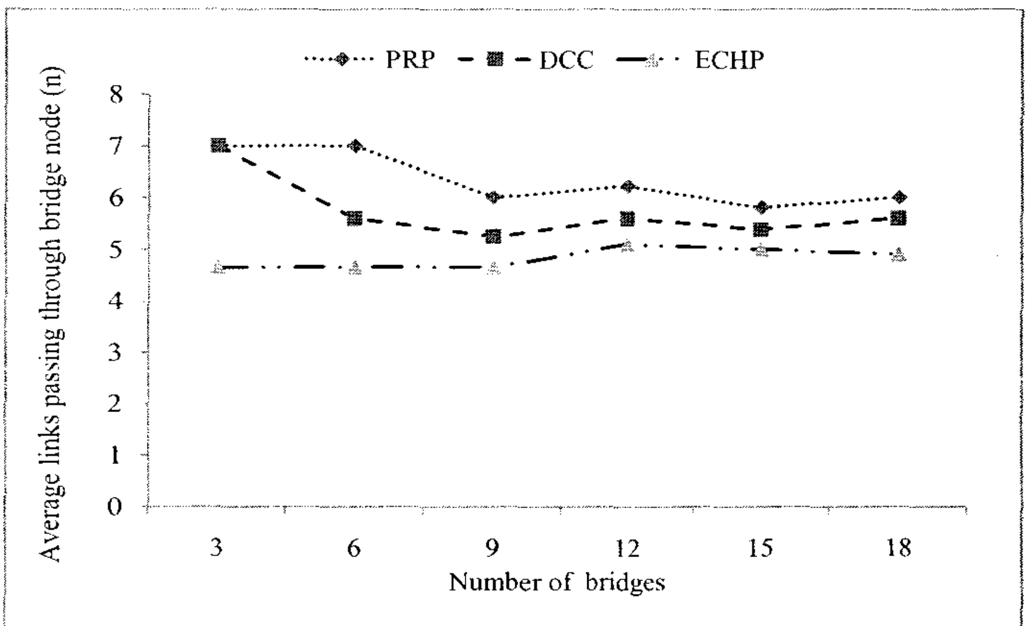


Figure 4.7: Average links passing through bridge nodes vs. number of bridges

4.2.2. An Efficient Route Maintenance Protocol for Dynamic Bluetooth networks

For route maintenance, an efficient route maintenance protocol, ERMP, has been proposed that repairs the weak routing links. This protocol is considered as efficient because it predicts the energy level and mobility of devices. During the transmission, the devices having critical energy levels can break the routing link anytime, whereas if a device starts moving from its original position, then there are chances for link breakage. So, the purpose of this protocol is to repair the routing link before its breakage. The ERMP was implemented and compared with the previous relevant protocols, ROMA and DENM. To evaluate the ERMP, DENM and ROMA protocols, the performance is measured from performance parameters as listed in Table 4.3. The protocols were compared in terms of the number of successfully repaired links, control overhead, energy consumption, data packet loss, healing delay and throughput. These parameters were chosen for route maintenance because the number of successful repaired links shows the overall repaired link, control overhead was chosen because it shows that during routing and route maintenance how much the control overhead occurred. The less energy consumption, less packet loss, minimum healing delay and higher throughput shows that how much the protocol is efficient.

Table 4.3: Performance parameters

Parameters	ERMP	DENM	ROMA
Successfully repaired links	√	√	√
Control overhead	√	√	√
Energy consumption	√	√	√
Data packet loss	√	√	√
Healing delay	√	√	√
Throughput	√	√	√

4.2.2.1. Successfully repaired links

The successfully repaired links demonstrated how many weak links/devices had been repaired. The proposed ERMP was compared with the previous DENM and ROMA protocols and it is observed from Fig.4.8, that the proposed ERMP successfully repaired more links as compared to the ROMA and DENM protocols. As the proposed ERMP predicted the weak links it activated a new link on run time before the main link broke. An example, shown in Fig.3.7b, shows how the ERMP repaired the weak routing links before the main route breakage. In this way, the ERMP overcame the problem of the extra resource utilization. On the other hand, ROMA did not provide a solution for the route recovery on run time when an active routing link broke or failed. It started to find a new link from the inquiry scan process which consumed more time. The DENM protocol just predicted the energy level of the devices. When it predicted a lower energy level, it activated a backup device but it did not provide any solution when a link broke because of mobility. Therefore, DENM has a lower number of successful route repairs as compared to ERMP. On the other hand, it can be observed that it repaired more failed links as compared to ROMA.

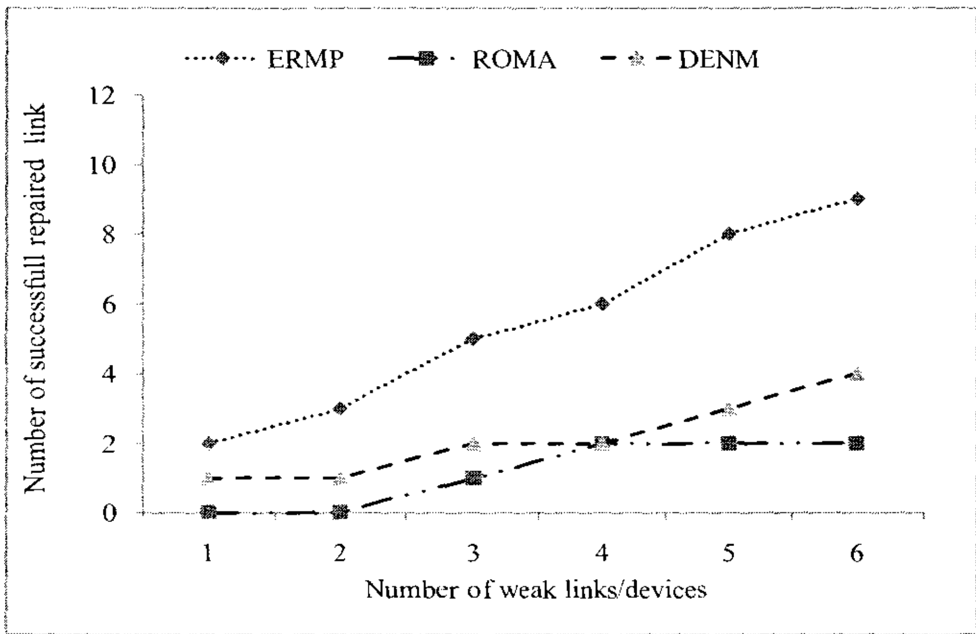


Figure 4.8: Number of successfully repaired links vs. Number of weak links/devices

4.2.2.2. Total control overhead

For synchronization, Bluetooth devices use control packets for network establishment. It has been analyzed that due to the limited battery power and mobility, Bluetooth frequently performs network restructuring and utilizes extra control packets that ultimately degrade the network performance. Therefore, if a critical battery level and mobility of a node is predicted, it can reduce frequent link disconnection that can ultimately reduce control overhead. From Fig.4.9, it is observed that the proposed ERMP has less control packet overhead compared to the DENM and ROMA protocols because both of these protocols do not perform network restructuring for route maintenance. Whereas, the proposed protocol keeps the information of the FBDs and whenever, a weak link because of mobility, or a weak device having a critical energy level is predicted, it activates a FBD for route maintenance and prevents the extra resource utilization. Although, ROMA is a route maintenance protocol, it changes the overall network structure for the reduction of hop counts when a new device joins or leaves the routing piconet. ROMA performs the extra resource utilization and creates the control overhead. It has been analyzed that the control packet overhead increased in all three protocols with the increase of the received packets.

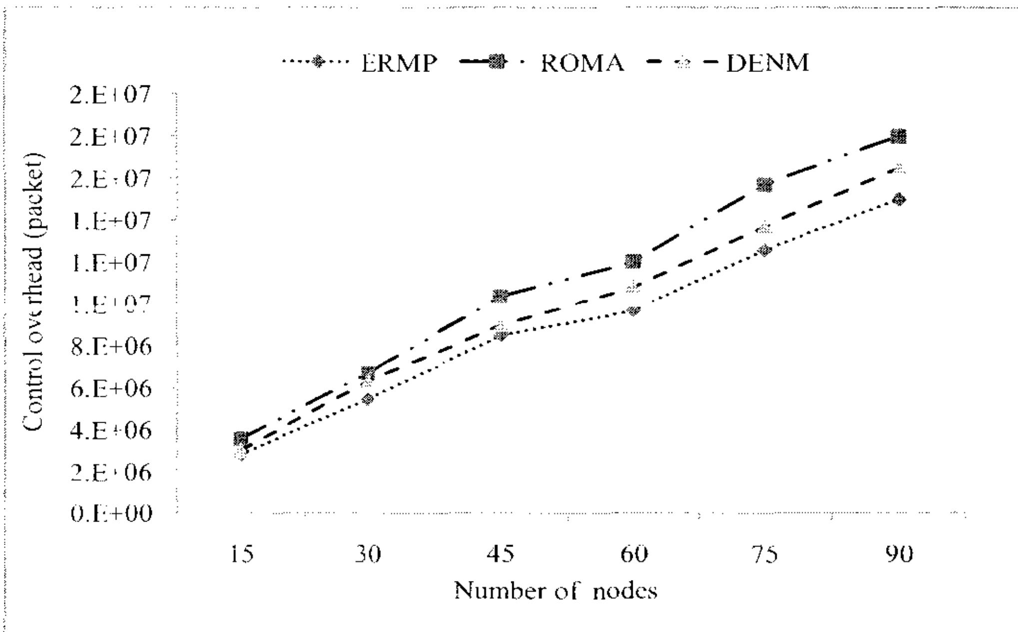


Figure 4.9: Control overhead vs. number of nodes

4.2.2.3. Total energy consumption

During the simulation, it was analyzed that when the data was routed between a source and a destination through the ROMA, each device updated its status to the connected master. When devices left or joined the network, they informed their respective master devices about their activities and waited for the reply. Therefore, ROMA consumed extra energy and other resources. On the other hand, the DENM protocol monitored the energy level of the devices. If it predicted a device having a critical energy level, it maintained a new link by activating a backup device. In case the link failed due to the mobility of any routing device, then this protocol did not show any action. As compared to the ROMA and DENM protocols, the ERMP consumed less energy because during the transmission it monitored the mobility and energy level of the devices. The master device replaced the device having a low energy level with the device having the maximum energy level to reduce the chance of the network partitioning. Fig.4.10 shows the energy consumption of the ERMP, ROMA and DENM protocols.

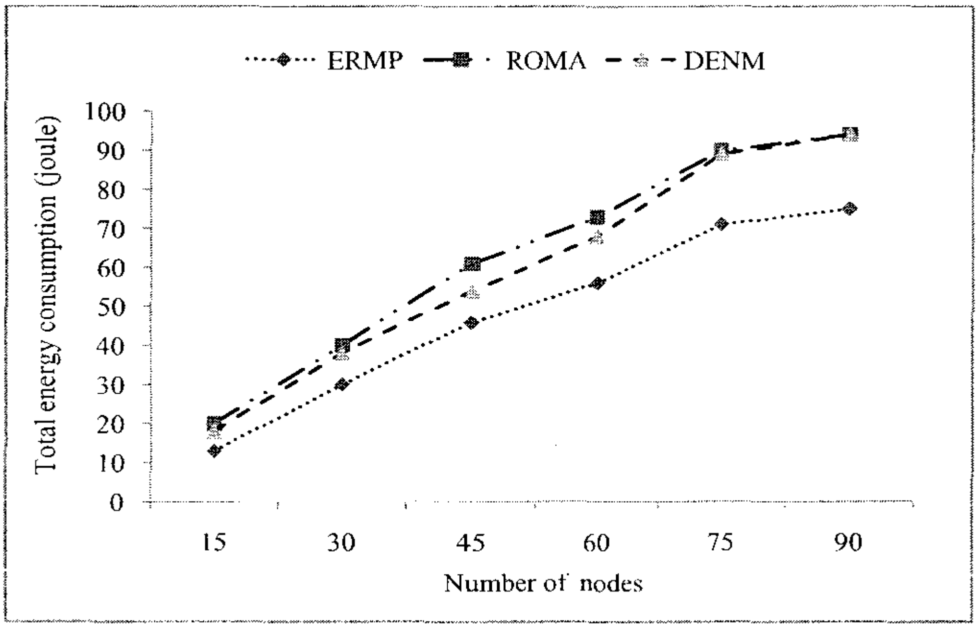


Figure 4.10: Total Energy consumption vs. number of nodes

4.2.2.4. Data packet loss percentage

Packet loss occurs, when data packets across the network fail to reach their final destination (Chih-Yung and Hsu-Ruey 2006). The main purpose of the proposed protocol is route maintenance which is based on the prediction of weak links. Therefore, it activates the alternate link as the weak main link is predicted. The proposed ERMP reduces data packet loss. From Fig.4.11, it is observed that the ROMA and DENM protocols lose more data packets. The reason is that, the DENM protocol provides the solution for route recovery if a low energy level of any device is predicted but it does not provide the solution if any routing device moves away and breaks the link. When devices are disconnected due to mobility between the main routing links, the DENM protocol loses more data packets. On the contrary, ROMA is also a route maintenance protocol but it does not provide the best solution for broken links. As during most of the situations, source and intermediate devices are unable to forward the data to the destination device because of the link failure but the ERMP protocol prevents the packet loss by maintaining new routes before links fail. From Fig.4.11, it has been analyzed that the proposed ERMP reduces the packet loss percentage compared to the previous DENM and ROMA protocols. It has also been observed that as the number of nodes increase it also increases the network size, which increases the path length. For the longer path, more numbers of intermediate devices are used, which increases the chances of link breakage. Therefore, it is observed that when the number of devices increases it also increases the packet loss in the network.

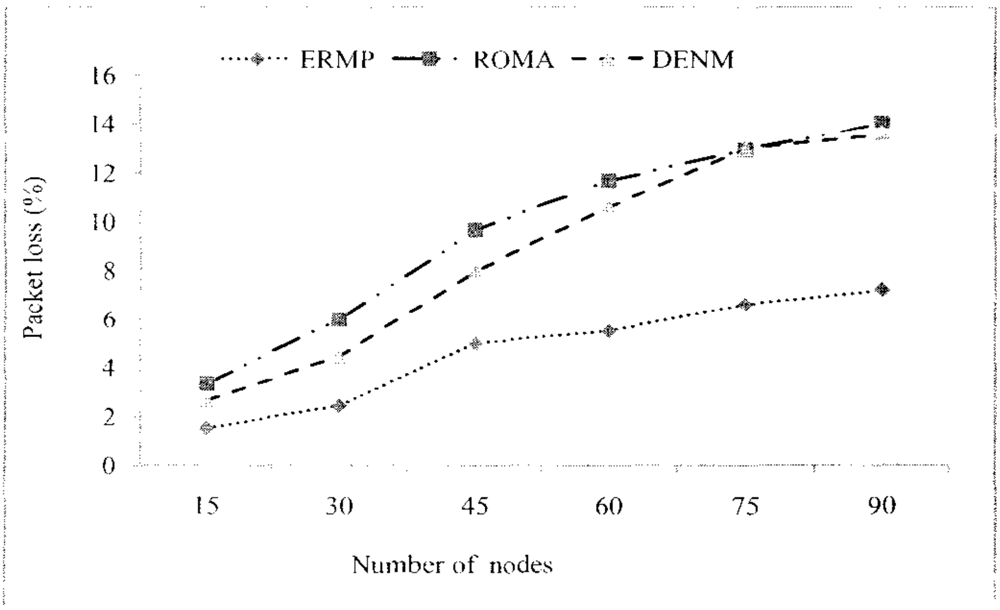


Figure 4.11: Packet loss vs. number of nodes

4.2.2.5. Healing delay

Fig.4.12 shows the healing delay of the protocols. After getting the simulation results, it was observed that the proposed ERMP utilized the minimum route recovery time as compared to the ROMA and DENM protocols. The reason behind this is that the proposed ERMP is based on the prediction of weak links and weak devices and it always keeps the information of the FBDs for the route maintenance on run time. The work of the ERMP has been explained by an example in Fig.3.8 which shows how it prevents the link breakage and maintains the routing links. When the ERMP repairs the weak links, it starts from the paging process that takes maximum time (2.56sec) for each link. The DENM protocol takes more time as compared to the ERMP because it predicts the devices based on the energy level; if a device having a critical energy level is predicted, it activates a backup device. When it repairs the links by activating the backup device, it starts from the paging process that takes 2.56 sec for each link. In case a link fails due to the mobility, then it starts to re-establish the link and performs the inquiry, inquiry scan, page and page scan processes; that takes time, accordingly. The inquiry and inquiry scan takes 10.24sec, whereas the paging process takes the maximum 2.56 sec for each link. Route recovery caused by mobility takes the total time for a link of (2.56 + 10.24) sec. Therefore, it takes more time than the ERMP. On the contrary, the ROMA protocol requires the maximum healing time for route maintenance because it maintains the routing link if any device joins or leaves the network, or for the device or link replacement; however, it does not perform any action if suddenly a link breaks or devices fail. It reconstructs the routing paths if the communication between the networks stops; it starts from the inquiry, inquiry scan and paging processes that takes (2.56 + 10.24) sec for each link. Therefore, the ROMA creates more healing delay than the DENM and the ERMP.

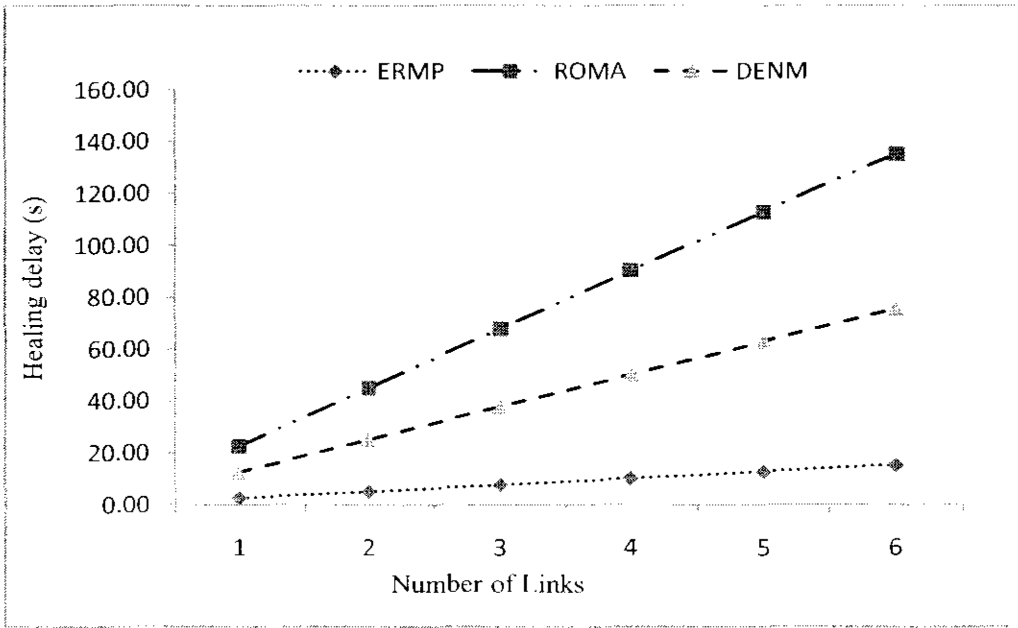


Figure 4.12: Healing delay vs. number of links

4.2.2.6. Network throughput

Throughput can be defined as the successful delivery of data packets per unit time across the network (Bong-Soo et al. 2004). It is observed from Fig.4.13 that the overall throughput of the proposed ERMP was higher than the ROMA and DENM protocols. During the simulation, the same number of data packets was routed across the network by using the ERMP, ROMA and DENM protocols. It was noted that the total number of received packets through the ERMP was more than the other protocols because it prevented the main routing link being broken and it always maintained the network transmission by predicting the status of the routing devices. Whereas, the throughput of the ROMA protocol was less than the proposed protocol, because once the main links broke it created all new links for the existing devices. Thus, ROMA increased the number of piconets which degraded the network performance. It was also analyzed that the throughput of the DENM protocol was also less than the ERMP. Due to the mobility; if the main link broke, it did not perform any action so it could not receive all of the data packets. Therefore, the proposed ERMP is considered as a more efficient protocol than the previous ROMA and DENM protocols.

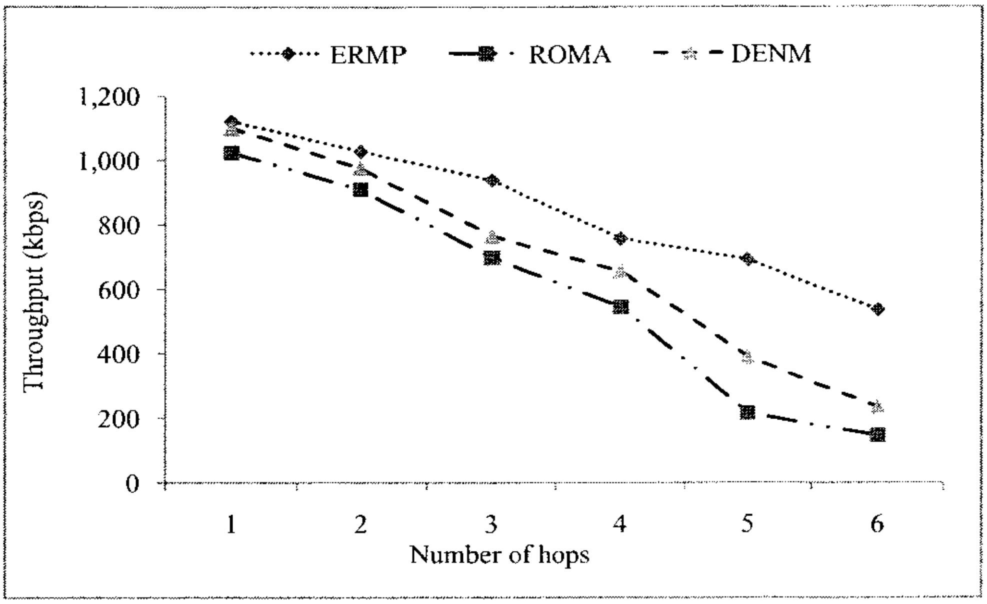


Figure 4.13: Network throughput vs. number of hops

4.2.3. An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network (CHRM)

In this sub-section, the comparison results of the new CHRM protocol are discussed. The results of this protocol have been compared with the results of the previous DCC and ROMA protocols. The DCC and ROMA protocols are selected for the comparison with CHRM because these protocols are advanced protocols of congestion handling and route maintenance. The CHRM protocol has the combined properties of the ECHP and ERMP protocols. The ECHP protocol handles the scatternet congestion and the ERMP gives an efficient solution for the route maintenance. Therefore, the CHRM is a combined protocol that gives the combined solution for the scatternet congestion and route maintenance. To assess the CHRM, DCC and ROMA protocols, the performance is measured from performance parameters as listed in Table 4.4. The protocols were compared in terms of the total delay, control overhead, network throughput, and data packet loss percentage. As, at the same time CHRM protocol monitors the energy level of devices, mobility and network congestion so these performance parameters were chosen to show that how much the protocol is efficient in terms of congestion handling and route maintenance compared to the other protocols.

Table 4.4: Performance parameters

Parameters	CHRM	DCC	ROMA
Total delay	√	√	√
Control overhead	√	√	√
Network throughput	√	√	√
Packet loss percentage	√	√	√

4.2.3.1. Total delay

The total delay of the CHRM was compared with the delay of the DCC and ROMA protocols. When the congestion arose on the master device, the DCC protocol handled the congestion by activating a backup device. The DCC protocol created more delay because when the congestion occurred on a bridge device, it did not activate backup devices. When one transmission ended, only then it allows the other transmission; so, during this process, it consumed more time. The ROMA protocol also created more delay because it did not provide the route recovery solution on run time. In this case, when an active link broke, then the ROMA protocol started to find a new link which consumed more time. The CHRM protocol consumed less time because if it predicted the traffic load on the master device or on the bridge device it performed the role switch operation. The CHRM protocol performed the role switch operation for most frequently communicating pairs. If it predicted that a single bridge device was providing multiple connections, then it activated the FBB to handle the load. During the transmission, the CHRM recovered the routes when it predicted the weak links and weak devices. It activated the FBDs before the main link breakage. The CHRM protocol took less time for the new connection activation which reduced the overall transmission delay. On the other hand, the ROMA protocol did not provide a solution for route recovery on run time. Once an active routing link broke, it started to find a new link which consumed more time as compared to the proposed protocol. From Fig.4.15, it was analyzed that the total delay of the CHRM was less than the previous DCC and ROMA protocols.

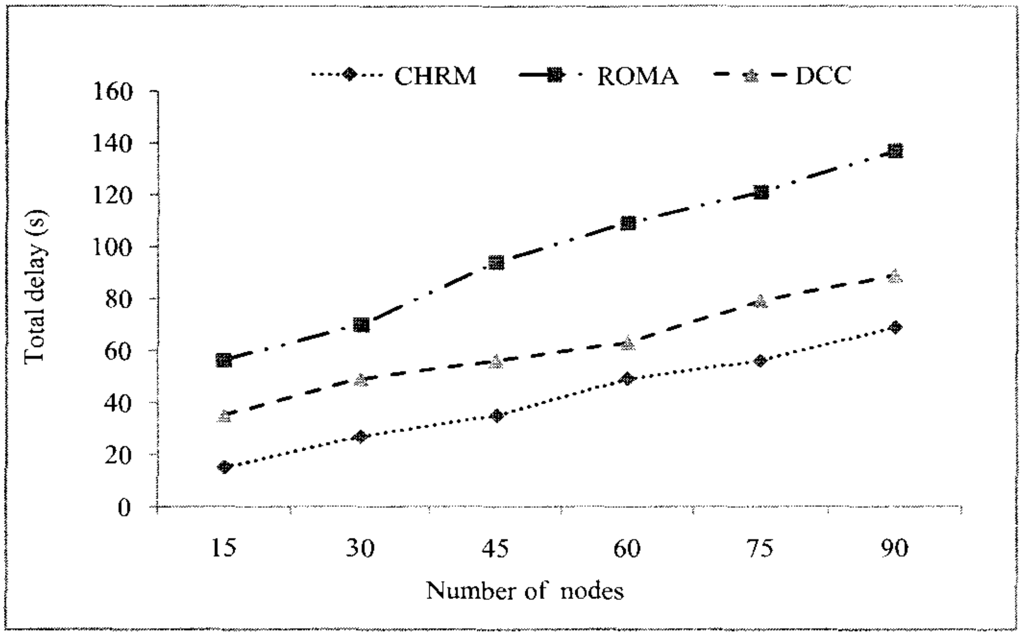


Figure 4.14: Total delay vs. number of nodes

4.2.3.2. Total control overhead

The control overhead of the CHRM was compared with the control overhead of the DCC and ROMA protocols. The previous DCC and ROMA protocols consumed more control packets as compared to the CHRM. The CHRM did not perform network restructuring very frequently; when congestion occurred, it only performed network restructuring for the most frequently communicating nodes. From Fig.4.15, it was observed that the control overhead of the CHRM was less than the DCC and ROMA protocols because both of the protocols did not provide a solution for the route maintenance on run time. Whereas, the proposed protocol kept the information of the FBDs, and whenever a weak link or weak device was predicted, it activated the FBD on run time and saved the extra resource utilization. It also prevented the link breakage. Although, the ROMA is a route maintenance protocol, it changed the overall network structure which created unnecessary control overhead. It can be analyzed that the control packet overhead increased in all three protocols with the increase of the received packets.

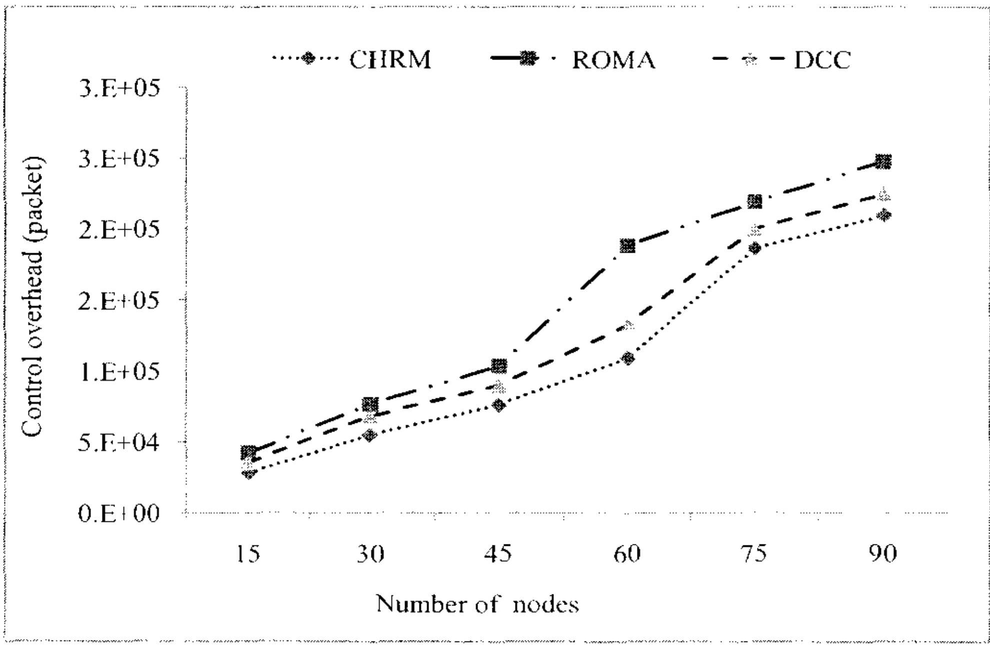


Figure 4.15: Control packet overhead vs. number of nodes

4.2.3.3. Network Throughput

It has been observed that the proposed CHRM increased the network throughput compared to the ROMA and DCC. The reason is that the CHRM always kept the information of the FBDs for the route maintenance. It repaired the link before the link breakage. By using this way, it prevented the extra resource utilization. On the other side, the ROMA protocol required more time for route maintenance because it maintained the routing links if any device joined or left the network, and for device or link replacement but it did not perform any action if a link suddenly broke or devices failed. It reconstructed the routing path if the communication between the networks stopped; therefore, the ROMA took a higher route recovery time. The DCC only monitored the traffic load on the master node and neglected the bridge node traffic load that increased delay. Fig.4.16 shows the throughput of the CHRM, DCC and ROMA protocols. From Fig.4.16, it was predicted that the throughput of the CHRM was higher than the DCC and ROMA.

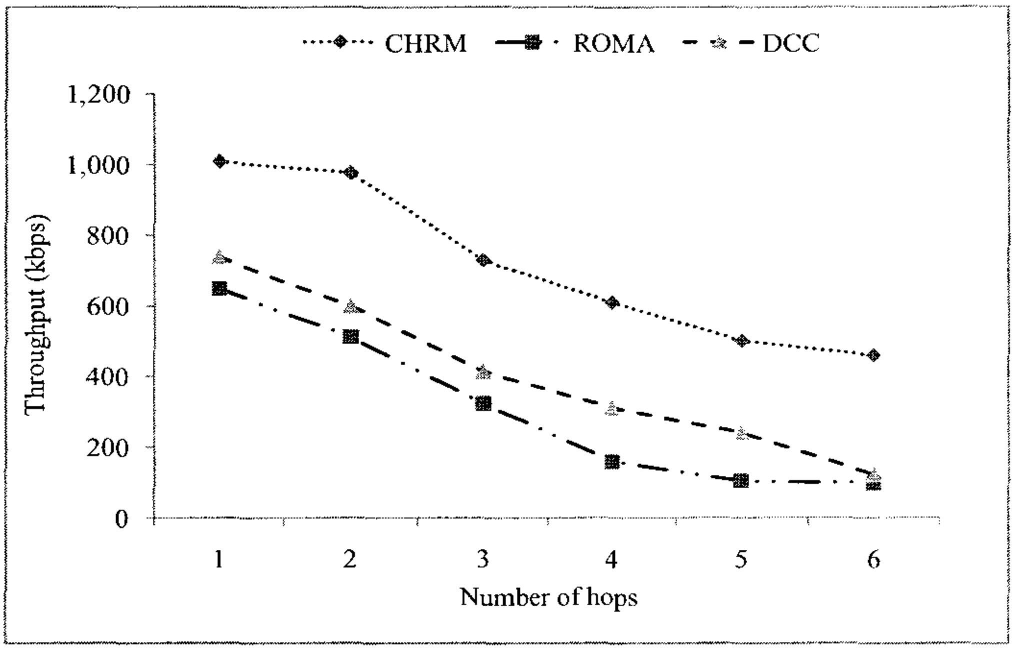


Figure 4.16: Throughput vs. number of hops

4.2.3.4. Packet loss percentage

The packet loss percentage of the CHRM was compared with the packet loss percentage of the DCC and ROMA protocols. In the case of the DCC protocol, the packet loss rate was higher because it did not support the congestion handling on the bridge node. ROMA is a route maintenance protocol but it did not provide the best solution for the broken links. It maintained a routing link only when a new device joined the piconet or left the piconet. During the transmission, most of the situations occurred when the source and intermediate devices were unable to forward the data to the destination device. Suppose an intermediate link between devices suddenly fails and the data is coming from the source device but an intermediate failed link is unable to forward the data to the destination device. During this situation, more data packets will not reach the destination. The CHRM protocol overcame the problem of data packet loss. It performed route maintenance based on the prediction of the energy level and the mobility of the devices. Therefore, it activated an alternate link as it predicted a weak routing link. By using this method, the proposed CHRM overcame the problem of data packet loss. From Fig.4.17, it is observed that the DCC and ROMA protocols lost more data packets compared to the CHRM.

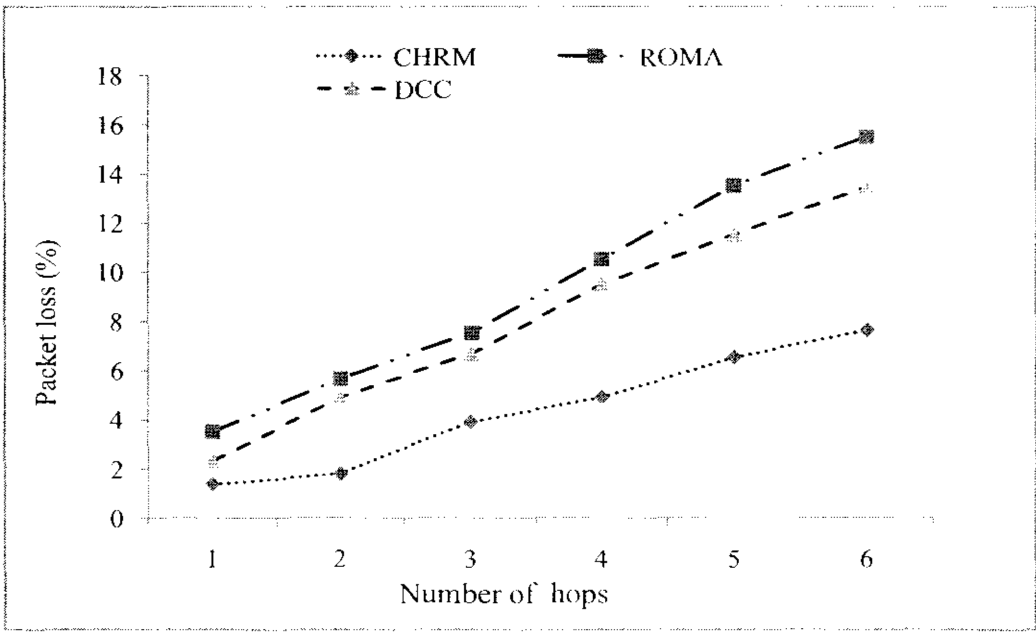


Figure 4.17: Packet loss vs. number of hops

4.3. SUMMARY

In this chapter, the results of the proposed methodology have been discussed. It described the following: 1) the simulator and simulation setup used for the implementation of the proposed protocol. 2) Comparison results of An Efficient Congestion Handling Protocol for Bluetooth scatternet, ECHP. It has been compared with the previous DCC and PRP protocols. 3) The comparison results of An Efficient Route Maintenance Protocol, ERMP. It has been compared with the previous ROMA and DENM protocols. 4) An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network CHRM. This protocol has been compared with the previous ROMA and DCC protocols. After getting the comparison results, it was observed that the proposed protocols outperform the previous protocols. The congestion handling ECHP considered the congestion on a master and bridge device at the same time and provides the efficient solution. The route maintenance ERMP protocol based on the prediction of mobility and energy level of devices; whenever a route breakage is predicted it activated its FBDs before the main route breakage. Therefore ERMP prevents the route breakage very efficiently. The third CHRM is a new protocol that combines the features of

ECHP and ERMP protocols. This protocol handles the problem of congestion and route maintenance in a single protocol.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This chapter emphasizes the conclusion of the "An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network". This is followed by the details of the achievement completed throughout this research work. The novelty of the research work is also presented. Lastly, the chapter discusses the future work.

5.1. CONCLUSION

In summary, this research work has developed An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network. This research work had three main parts. The first part of the research work contained An Efficient Congestion Handling Protocol for Bluetooth scatternet (ECHP). An inefficient Bluetooth topology may create a bottleneck and a delay in the network when data is routed. As the PRP frequently performs network restructuring which creates more control overhead. In contrary, the DCC protocol does not consider real network restructuring so it depends on a backup node, it has been observed that in many cases the backup node is hard to find. Moreover, the DCC protocol uses the centralized approach; therefore, it does not truly identify the congestion on a bridge node. The proposed ECHP protocol handles the traffic load on the master node by network restructuring and handles the traffic load on the bridge node by creating a FBB. During the piconet restructuring, the ECHP protocol performs the PFP and SFP. The PFP reconstructs a new piconet in the same piconet for the source and destination which are directly within the radio range of each other, and the SFP reconstructs the scatternet within the same piconet if the source and destination nodes are not within the radio range of each other. If congestion occurs on a bridge device, the ECHP protocol creates a FBB within the scatternet that handles the congestion on the bridge node and provides an efficient way for parallel transmission. Analytically, the ECHP has efficiently handled the network traffic load and improved the network performance.

The second part consisted of an Efficient Route Maintenance Protocol (ERMP) for the dynamic Bluetooth network. Initially, an active and connected Bluetooth scatternet was considered and maintained the routing links. After the main route construction, each routing master and routing bridge device kept the information of all of the connected devices. The proposed protocol predicted the weak links and weak devices through the signal strength and energy level of the devices and repaired the weak links by activating the FBDs. It was analyzed that the ERMP was more efficient than the ROMA and DENM protocols because both of the protocols provided a solution for route breakage only when the main route had been broken. On the contrary, the proposed ERMP predicted the weak routing links and weak devices, and if a weak routing link was predicted then this protocol made a new link before the main route broke. Through the simulation results, it was observed that the proposed ERMP performed route maintenance, efficiently.

The third part of the research work consisted of An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network (CHRM). The new CHRM protocol is a combination of both of the proposed protocols, ECHP and ERMP, into one protocol. The CHRM protocol is an efficient protocol that has been observed to reduce the congestion in a scatternet and it also performs the route maintenance for weak links. It has been seen to repair the weak routing links based on the prediction of the energy level and mobility of the Bluetooth devices. This protocol has been implemented and compared with the previous ROMA and DCC protocols. From the comparison results it has been observed that the CHRM protocol performed better in terms of total delay, control overhead, packet loss, route recovery time etc.

5.2. CONTRIBUTION AND NOVELTY

The contribution of this research work is significant for Bluetooth technology but it is more helpful as it provides the solution for the scatternet congestion handling and route maintenance.

The proposed An Efficient Combined Congestion Handling and Route Maintenance Protocol for Dynamic Environment in Bluetooth Network is an efficient protocol that handles the network congestion and provides the optimum solution for route maintenance. During transmission, when congestion occurs on a master device, it performs piconet restructuring by using role switch operations. If congestion occurs on a bridge device as the cause of heavy data traffic load from multiple connected piconets, then the proposed protocol activates a FBD that shares the congestion as well as allowing for the parallel transmission. During the transmission, if it predicts devices having critical energy levels such that can break the routing link or if it predicts that a device is moving away, which can also break the routing link, then it activates alternate devices to recover the link before it breaks. The proposed protocols have been implemented and it has been observed from the simulation results that this protocol outperforms the other protocols.

The main novelty is listed below:

i. Our protocol efficiently handles the congestion and problem of route breakage. Therefore it minimises the healing delay, as it predicts the traffic load on the master device or on the bridge device; then it performs the role switch operation only for most frequently communicating pairs. If more piconets are communicating through a single bridge device, then it activates the FBB to share the load. During transmission, it recovers the route when it predicts a device having a critical energy level or mobility. It activates the FBDs before the main link breakage. It takes less time for new connection activation which reduces the overall transmission delay.

ii. It reduces the control packet overhead. For congestion handling, it does not perform network restructuring very frequently, only when the congestion occurs then it executes the network restructuring only for the most frequently communicating nodes. It updates the table and keeps the list of FBDs for route maintenance. It prevents the link breakage so it does not allow the extra resource utilisation. For route maintenance, our protocol predicts the weak links and weak devices when any weak link or weak device is predicted it immediately activated FBD to prevent the link breakage.

iii. The other novelty of this research work is that it repairs the maximum links, successfully. As it predicts a weak link, it activates a FBD. In case there is no alternate device available, then this protocol performs the role switch operation.

iv. The CHRM protocol is developed which is the first protocol that provides the solution for congestion handling and route maintenance in a same protocol. This protocol was implemented because during transmission some time when congestion is handled then problem of route maintenance occurs and when sometimes route maintenance is handled then the problem of congestion occurs. Therefore we implemented a new protocol that considers the both problems of congestion and route maintenance at the same time and provide solution.

v. After getting the simulation results, it was observed that ECHP, ERMP and CHRM increased the network throughput by receiving more data packets. Within the scatternet, parallel transmission can increase the network performance; therefore, it allowed the parallel transmission when a single bridge device was providing multiple connections to multiple piconets. It also increased the throughput by preventing the link breakage. Therefore, it always maintained the network transmission by predicting the status of the routing devices.

5.3. FUTURE WORK

The future directions of this research work are given below:

1. As the proposed protocol was implemented in a dynamic environment. In future, Heterogeneous technology (TV White Space) can be used in a static environment to check the network performance.
2. The proposed protocol shares the congestion of the master device and bridge device by network restructuring or role switch operations. In future network, congestion can be shared by reducing the hop count based on the role switch operations.
3. It performs route maintenance by predicting the mobility and energy consumption of devices. In future, routes can be maintained by allocating time slots to each route for communication based on stable nodes.

REFERENCES

The Network Simulator ns-2, http://www.isi.edu/nsnam/ns/ns-build.html, July 2011.

University of Cinicinnati Bluetooth simulator (UCBT), http://www.ececs.uc.edu/_cdmc /ucbt/>, June 2011.

Al-Jarrah O, and Megdadi O, "Enhanced AODV routing protocol for Bluetooth scatternet", Elsevier Journal: Computers & Electrical Engineering, Vol. 35(1), pp: 197-208, 2009.

Alkhrabash A.A.S, and Elshebani M, "Routing schemes for Bluetooth scatternet applicable to mobile ad-hoc networks", In proc of: TELSIKS 9th International Conference on: Telecommunication in Modern Satellite, Cable, and Broadcasting Services, pp: 560-563, 2009.

Bakhsh S.T, Hasbullah H, Tahir S, and Subhan F, "Dynamic energy-aware network maintenance for Bluetooth", In proc of IEEE: International Conference of Information Science and Applications, pp: 1-9, 2011.

Basagni S, Bruno R, Mambrini G, and Petrioli C, "Comparative Performance Evaluation of Scatternet Formation Protocols for Networks of Bluetooth Devices", Journal of Wireless Networks Vol. 10(2), pp: 197-213, 2004.

Blom M, Ekstrom M, Castano J.G, and Linden M, "Bluetooth energy characteristics in wireless sensor networks" In proc of: ISWPC 3rd International Symposium on Wireless Pervasive Computing, pp: 198-202, 2008.

Bluetooth_specificaiton, Bluetooth SIG, "Specification of the Bluetooth System", http://www.bluetooth.com, 2010.

Bong S.L, H W.K, Tae U.Y, Yong S.P, Byeong H.A, and Joung C.A, "A new efficient scheduling algorithm in Bluetooth piconet", In proc of IEEE: Global Telecommunications Conference, GLOBECOM '04, Vol. 3606, pp: 3609-3613, 2004.

Petrioli C, Stefano B, and Iinrich C, "Configuring bluestars: Multi-hop scatternet formation for bluetooth networks". Journal of IEEE Transactions on Computers: Special Issue on Wireless Internet, Vol. 52(6) pp: 779–790, 2003.

Chakrabarti S, Liyun W, Son V, and Leung, "VCMA remotely controlled Bluetooth enabled environment", In proc of IEEE: Consumer Communications and Networking Conference, CCNC, pp: 77-81, 2004.

Chakraborty G, Naik K, Chakraborty D, Shiratori N, and Wei D, "Analysis of the Bluetooth device discovery protocol", Springer Journal: Wireless Networks, Vol. 16(2), pp: 421-436, 2010.

Chang C.T, Chang C.Y, and Chang S.W, "TMCP: Two-layer multicast communication protocol for Bluetooth radio networks", Elsevier Journal: Computer Networks Vol. 52(14), pp: 2764-2778, 2008.

Chang C.T, Chang C.Y, and Sheu J.P, "BlueCube: Constructing a hypercube parallel computing and communication environment over Bluetooth radio systems", Elsevier Journal: Parallel and Distributed Computing, Vol. 66(10), pp: 1243-1258, 2006.

Chang S.W, Sahoo P, and Chang C.Y, "A Location Aware Mobility based Routing Protocol for the Bluetooth Scatternet", Springer Journal: Wireless Personal Communications Vol. 47(4), pp: 541-566, 2008.

Chaudhry M.J, Murawwat S, Saleemi F, Tariq S, Saleemi M, and Chaudhry FJ, "Power optimized secure Bluetooth communication", In proc of IEEE: Multi-topic Conference, INMIC International, pp: 182-188, 2008.

Chen Y.S, Lin Y.W, and Chang C.Y, "An overlapping communication protocol using improved time-slot leasing for Bluetooth WPANs", Elsevier Journal: Network and Computer Applications Vol. 32(1), pp: 273-292, 2009.

Chen Y.S and Liu K.S, "A Credit-Based On-Demand QoS Routing Protocol over Bluetooth WPANs", Springer Journal: Wireless Personal Communications, Vol. 38(2), pp: 253-278, 2006.

Chen Z, Jens E, Rumen K, and Delsing. J, "Evaluation of a Heterogeneous Sensor Network Architecture for Highly Mobile Users", Journal of Scientific Research: Communications and Network, Vol. (3), pp: 57-64, 2011.

Chih M.Y, and Yi W.C, "HMT: A hybrid mesh tree approach in forming Bluetooth networks", In the proc of IEEE: Computer Symposium (ICS) International: pp: 441-444, 2010.

Chih Y.C, and Hsu R.C, "Adaptive role switching protocol for improving scatternet performance in Bluetooth radio networks", Journal of IEEE Transactions: Consumer Electronics, Vol. 52(4), pp: 1229-1238, 2006.

Ching F.H, and Shu M.H, "An Adaptive Inter-piconet Scheduling Algorithm Based on HOLD Mode in Bluetooth Scatternets", Journal of IEEE Transactions: Vehicular Technology, Vol. 57(1), pp: 475-489, 2008.

Ching C.C, Prasan K.H, and Lee S.C, "A Location-Aware Routing Protocol for the Bluetooth Scatternet", Springer Journal: Wireless Personal Communications, Vol. 40(1), pp: 117 – 135, 2007.

Chung H.L, and Yun M.H, "The Study of the Bluetooth Column-Ring path", In proc of: Fifth International Joint Conference on: INC, IMS and IDC, pp: 1552-1555, 2009.

Cordeiro C.D.M, Abhyankar S, Toshiwal R, and Agrawal D.P, "A novel architecture and coexistence method to provide global access to/from Bluetooth WPANs by IEEE 802.11 WLANs", In proc of IEEE: International conference on Performance, Computing, and Communications Conference: pp: 23-30, 2003.

Cui Y.d, Xiao Z.q, Wang X.y, and Yang B.x, "Bluetooth energy-saving optimization: a queuing theory analysis", The Journal of China Universities of Posts and Telecommunications 17(Supplement 1): pp: 54-57, 2010.

De Morais C.C, Abhyankar S, and Agrawal D.P, "Reducing power consumption and enhancing performance by direct slave-to-slave and group communication in Bluetooth WPANs", Journal of IEEE: Computer Networks, Vol. 45(2), pp: 119-141, 2004.

Dunne, K, Elaine R, David L, and Rhatigan E, "Bluetooth for Ad-Hoc Networking", NTRG: Networks & Telecommunications Research Group Networks and Telecommunications Research Group (NTRG), Trinity College Dublin, Ireland, 2005.

Emre A, and Oznur O, "A Classification and Performance Comparison of Mobility Models for Ad Hoc Networks", In proc of Springer: ADHOC-NOW Istanbul, Turkey, pp: 444 – 457, 2006.

Gonzalez C.F.J, and Garcia R.J, "Survivable Bluetooth Location Networks" In proc of IEEE: International Conference on Communications ICC, pp: 1014-1018, 2003.

Gelzayd Y, "An Alternate Connection Establishment Scheme in the Bluetooth System" [Degree of Master of Science (Telecommunication Networks)]: Polytechnic University, 2002.

Hassan T, Kayssi A, and Chehab A, "Ring of Masters (ROM): A new ring structure for Bluetooth scatternets with dynamic routing and adaptive scheduling schemes", Elsevier Journal: Pervasive and Mobile Computing, Vol. 4(4), pp: 546-561, 2008.

Jingli L, Xiangqian L, and Swami A, "Collision analysis for coexistence of multiple bluetooth piconets and WLAN with dual channel transmission", Journal of IEEE Transactions: Communications: Vol. 57(4), pp: 1129-1139, 2009.

Johansson P, Kapoor R, Kazantzidis A, and Gerla M, "Rendezvous scheduling in Bluetooth scatternets", In proc of IEEE: International Conference on Communications ICC: pp: 318-324, 2002.

Jong M.C, Istepanian R.S.H, Alesanco A, and Wang H, "Hardware design & compression issues in compact Bluetooth enabled wireless telecardiology system", In proc of IEEE: International Conference on BroadNets and Broadband Networks, Vol. 1012, pp: 1014-1015, 2005.

Jung S, Chang A, and Gerla M, "New Bluetooth interconnection methods: Overlaid Bluetooth Piconets (OBP) and Temporary Scatternets (TS)", Elsevier Journal: Computer Communications: Vol. 30(10), pp: 2258-2273, 2007.

Jung S, Chang A, and Gerla M, "Peer to peer video streaming in Bluetooth overlays", Springer Journal: Multimedia Tools and Applications, Vol. 37(3), pp: 263-292, 2008.

Jung S, Lee U, Chang A, Cho D-K, and Gerla M, "BlueTorrent: Cooperative content sharing for Bluetooth users", In proc of IEEE: Pervasive Computing and Mobile Communications, pp: 47-56, 2007.

Ka-W.H, and Luong H.C, "A 1-V CMOS power amplifier for Bluetooth applications", IEEE Journal of IEEE Transactions: Circuits and Systems II Analog and Digital Signal Processing, Vol. 50(8), pp: 445-449, 2003.

Kapoor R, Zanella A, and Gerla M, "A Fair and Traffic Dependent Scheduling Algorithm for Bluetooth Scatternets", Journal: Mobile Networks and Applications, Vol. 9(1), pp: 9-20, 2004.

Kazemian H.B, and Li M, "An adaptive control for video transmission over Bluetooth", Journal IEEE Transactions: Fuzzy Systems, Vol. 14(2), pp: 263-274, 2006.

Kazemian H.B, and Meng L, "A fuzzy control scheme for video transmission in Bluetooth wireless", Elsevier Journal: Information Sciences, Vol. 176(9), pp: 1266-1289, 2006.

Khan M.K, Tabassam A.A, and Ali F.A, "Simulation analysis of Bluetooth Piconets self-disturbance in industrial applications: A case study", In proc of ICUMT: International Conference on Ultra Modern Telecommunications & Workshops, pp: 1-8, 2009.

Khoutaif T, and Juanole G, "Formal _modeling and evaluation of the data transfer phase of the ACL links on the WPAN Bluetooth", In proc of IEEE: Emerging Technologies and Factory Automation, ETFA, pp: 30-37, 2006.

Kiss K.C, Chiasserini C.F, Jung S, Brunato M, and Gerla M, "Hop count based optimization of Bluetooth scatternets, Elsevier Journal: Ad Hoc Networks, Vol. 5(3), pp: 340-359, 2007.

Klajbor T, Rak J, and Wozniak J, "A new role-switching mechanism optimizing the coexistence of Bluetooth and Wi-Fi networks", Springer Journal: Telecommunication Systems, Vol. 48(3-4), pp: 277-287, 2011.

Klajbor T, and Woźniak J, "Interference aware Bluetooth scatternet (re)configuration algorithm IBLUEREA", Springer Journal: Telecommunication Systems, Vol. 38(1), pp: 29-35, 2008.

Kleinschmidt J.H, Borelli W.C, and Pellenz M.E, "An energy efficiency model for adaptive and custom error control schemes in Bluetooth sensor networks AEU", Elsevier Journal: Electronics and Communications, Vol. 63(3), pp: 188-199, 2009.

Le S.N, and Ben S.S, "Location-Based Service Discovery and Delivery in Opportunistic Networks. Networks (ICN)", In proc of IEEE Computer Society: Ninth International Conference on, pp: 179-184, 2010.

Lee S.H, and Lee Y.H, "Adaptive frequency hopping and power control based on spectrum characteristic of error sources in Bluetooth systems", Elsevier Journal: Computers & Electrical Engineering, Vol. 36(2), pp: 341-351, 2010.

Leopold M, Dydensborg M, and Bonnet P, "Bluetooth and sensor networks: a reality check", In the proc of ACM: 1st International Conference on Embedded Networked Sensor Systems, SenSys, Los Angeles, USA, 2003.

Lin J.W, and Wang W.S. "An efficient reconstruction approach for improving Bluetree scatternet formation in personal area networks", Elsevier Journal: Network and Computer Applications, Vol. 33(2), pp: 141-155, 2010.

Lin T.Y, Tseng Y.C, and Chang K.M, "A new BlueRing scatternet topology for Bluetooth with its formation, routing, and maintenance protocols", Journal: Wireless Communications and Mobile Computing, Vol. 3(4), pp: 517-537, 2003.

Lo B.L, and Mirabella O, "Clock synchronization issues in Bluetooth-based industrial measurements". In proc of IEEE: International Workshop on Factory Communication Systems, pp: 193-202, 2006.

McDermott-Wells P, "What is Bluetooth?" IEEE Journal Potentials, Vol. 23(5), pp: 33-35, 2004.

Mei, Jiexiang M, Salim, and Agus, "Access Control using Bluetooth TM09", Sydney, Australia: THE UNIVERSITY OF NEW SOUTH WALES SCHOOL, 2003.

Misic J, Reddy G.R, and Misic V.B, "Activity scheduling based on cross-layer information in Bluetooth sensor networks", Elsevier Journal: Computer Communications, Vol. 29(17), pp: 3385-3396, 2006.

Mišić VB, and Mišić J, "Performance of Bluetooth Bridges in Scatternets with Limited Service Scheduling", Springer Journal: Mobile Networks and Applications, Vol. 9(1), pp: 73-87, 2004.

Moron M, Luque R, Casilari E, and Diaz E.A, "Overhead and Segmentation Mismatch Effect on Bluetooth WPAN Performance", Springer Journal: Wireless Personal Communications Vol. 50(2), pp: 161-180, 2009.

Movahhedinia N, and Ghahfarokhi B.S, "Performance analysis of Bluetooth asynchronous connection-less service", Elsevier Journal: Network and Computer Applications, Vol. 34(2), pp: 731-738, 2011.

Naik K, Wei, Su Y.T, and Shiratori N, "Analysis of packet interference and aggregated throughput in a cluster of Bluetooth piconets under different traffic conditions", IEEE Journal: Selected Areas in Communications, Vol. 23(6), pp: 1205-1218, 2005.

Naik K, Wei, Su Y.T, and Shiratori N, "A random graph-based model to analyze packet interference between frequency hopping systems with an application to Bluetooth", Elsevier Journal: Computer Communications", Vol. 31(14), pp: 3286-3291, 2008.

Ophir L, Bitran Y, and Sherman I, "Wi-Fi (IEEE 802.11) and Bluetooth coexistence: issues and solutions", In proc of IEEE: International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 15th, Vol.842, pp: 847-852, 2004.

Pasolini G, and Verdone R, "Analytical Evaluation of Throughput for a Bluetooth Piconet with MAC Level Link Adaptation", Springer Journal: Mobile Networks and Applications, Vol. 10(5), pp: 717-725, 2005.

Persson K.E, Manivannan D, and Singhal M, "Bluetooth scatternets: criteria, models and classification", Journal of IEEE: Ad Hoc Networks, Vol. 3(6), pp: 777-794, 2005.

Peters A, and Heuer A, "Time Efficient Service Discovery in Mobile P2P Architectures Using Bluetooth", In proc of IEEE Computer Society: Database and Expert Systems Application, DEXA '08 19th International Workshop on, pp: 347-351, 2008.

Petrioli C, Basagni S, and Chlamtac I, "Configuring BlueStars: Multihop scatternet formation for Bluetooth networks", In proc of IEEE Trans Computers: pp: 779-790, 2003.

Ramamurthy H, Prabhu B.S, Gadh R, and Madni A.M, "Wireless Industrial Monitoring and Control Using a Smart Sensor Platform", Journal of IEEE: Sensors, Vol. 7(5), pp: 611-618, 2007.

Ramana Reddy G, Bhatnagar S, Rakesh V, and Chaturvedi V, "An efficient algorithm for scheduling in Bluetooth piconets and scatternets", Journal Wireless Networks: Vol. 16(7), pp: 1799-1816, 2010.

Roger M, Whitaker, Leigh H, and Chlamtac. I, "Bluetooth scatternet formation: A survey", Elsevier Journal: Ad Hoc Networks, pp: 403-450, 2005.

Rotaru M, Lim Y.Y, Kuruveettil H, Yang R, Popov A.P, and Chua C.P, "Implementation of Packaged Integrated Antenna With Embedded Front End for Bluetooth Applications", Journal of IEEE Transactions on : Advanced Packaging, Vol. 31(3), pp: 558-567, 2008.

Sahoo PK, Chang C.Y, and Chang S.W, "Novel route maintenance protocols for the Bluetooth ad hoc network with mobility", Elsevier Journal: Network and Computer Applications, Vol. 31(4), pp: 535-558, 2008.

Salonidis T, Bhagwat P, Tassiulas L, and LaMaire R, "Distributed topology construction of Bluetooth personal area networks", In proc of IEEE: Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM, pp: 1577-1586, 2001.

Salonidis T, Bhagwat P, Tassiulas L, and LaMaire R, "Distributed topology construction of Bluetooth wireless personal area networks", Journal of IEEE: Selected Areas in Communications, Vol. 23(3), pp: 633-643, 2005.

Shek L.Y, and Kwok Y.K, "An integrated approach to scatternet traffic management in Bluetooth ad hoc networks", Elsevier Journal: Computer Networks, Vol. 45(2), pp: 99-118, 2004.

Sheng W.C, Sahoo P.K, Li L.H, and Chih Y.C, "A location-and-mobility aware routing protocol for Bluetooth radio networks", In proc of IEEE: Joint Conferences on Pervasive Computing JCPC, pp: 137-142, 2009.

Shih C.H, Wang K, and Shih H.C, "An adaptive Bluetooth packet selection and scheduling scheme in interference environments", Elsevier Journal: Computer Communications, Vol. 29(11), pp: 2084-2095, 2006.

Sreenivas H, and Ali H, "An Evolutionary Bluetooth Scatternet Formation Protocol", In the Proc of IEEE: 37th Hawaii Int Confon System Sciences, 2004.

Stojmenovic I, and Zaguia N, "Bluetooth scatternet formation in ad hoc wireless networks", Power-aware localized routing in wireless networks. University of Ottawa, Ontario K1N 6N5, Canada, 2001.

Bakhsh S.T, Hasbullah H, and Tahir S, "Dynamic Congestion Control through backup relay in Bluetooth scatternet", Elsevier Journal: Network and Computer Applications, Vol. 34(4), pp: 1252-1262, 2011.

Teerawat I, and Ekram H, "Introduction to Network Simulator NS 2". University of Manitoba, Canada, 2009.

Tekkalmaz M, Sozer H, and Korpeoglu I, "Distributed Construction and Maintenance of Bandwidth and Energy Efficient Bluetooth Scatternets", Journal of IEEE Transactions on: Parallel and Distributed Systems, Vol. 17(9), pp: 963-974, 2006.

Vergetis E, Guerin R, Sarkar S, and Rank J, "Can Bluetooth succeed as a large-scale ad hoc networking technology?", Journal of IEEE: Selected Areas in Communications, Vol. 23(3), pp: 644-656, 2005.

Vojislav B, Misic., and J. M, "Polling and Bridge Scheduling Algorithms in Bluetooth", Technical report TR 03/04 Department of Computer Science, University of Manitoba Winnipeg, Manitoba, Canada R3T 2N2, 2003.

Wang Z, Thomas R, and Haas Z, "Performance comparison of Bluetooth scatternet formation protocols for multi-hop networks", Springer Journal: Wireless Networks, Vol. 15(2), pp: 209-226, 2009.

Wei Y.H, Jia W.L, Kuo C.T, Yong H.H, Chao L.C, Hung Ta.T, Yi S.S, Shao C.H, and Sao J.C, "A CMOS RF Transceiver With Self-Detection and Calibration Functions for Bluetooth V2.1 EDR Applications", Journal of IEEE Transactions: Microwave Theory and Techniques, , Vol. 58(5), pp: 1367-1374, 2010.

Whitaker R.M, Hodge L, and Chlamtac I, "Bluetooth scatternet formation: A survey", Elsevier Journal: Ad Hoc Networks, Vol. 3(4), pp: 403-450, 2005.

Xiang L, Chaoguang M, Mantian L, and Lining S, "Hierarchical Routing for Large Scale Bluetooth Network", In the proc of IEEE: Networks Security, Wireless Communications and Trusted Computing, NSWCTC International Conference on, pp: 415-418, 2009.

Xin Y, and Yu P.W, "Secure Constructing Bluetooth Scatternet Based on BTCP", In the proc of IEEE Fifth International Conference on: Information Assurance and Security, IAS, pp: 200-203, 2009.

Yip H.K, and Kwok Y.K, "Practical channel state aware and cooperative packet scheduling disciplines for coordinating collocated Bluetooth and IEEE 802.11b devices", Elsevier Journal: Computer Communications, Vol. 30(7), pp: 1569-1587, 2007.

Yu C, Yu K.M, and Lin S, "Efficient Scheduling Algorithms for Bluetooth Scatternets" Springer Journal: Wireless Personal Communications, Vol. 48(2), pp: 291-309, 2009.

Yu G.J, and Chang C.Y, "Congestion control of Bluetooth radio system by piconet restructuring", Elsevier Journal: Network and Computer Applications", Vol. 31(3), pp: 201-223, 2008.

Yu G.J, Chang C.Y, Shih K.P, and Lee S.C, "Relay reduction and disjoint routes construction for scatternet over Bluetooth radio system", Elsevier Journal: Network and Computer Applications, Vol. 30(2), pp: 728-749, 2007.

Yunbo W, Goddard S, and Perez L.C, "A study on the cricket location-support system communication protocols", In the proc of IEEE: International Conference on Electro/Information Technology, pp: 257-262, 2007.

Yu C. W, Yu K.M, Lin S.L, "Efficient Scheduling Algorithms for Bluetooth Scatternets", Springer Journal, Wireless Personal Communications, Vol. 48, pp: 291-309, 2009.

Shafie. A, "The Blue Zone –Mobile Bluetooth Network Expansion", In the proc of IEEE, 4th International Conference on Information and Automation for Sustainability, ICIAFS, pp: 407-412, 2008.

Tekkalmaz M, Sozer H, Korpeoglu I. "Distributed construction and maintenance of bandwidth and energy efficient bluetooth scatternets", IEEE Transactions Journal: Parallel and Distributed Systems, Vol. 17(9), pp: 963–74, 2006.

Lin J.W, Wang W.S, "An efficient reconstruction approach for improving Bluetree scatternet formation in personal area networks", Elsevier Journal: Network and Computer Applications, Vol. 33(2), pp: 141-155, 2010.

PUBLICATIONS

1. "A Bluetooth Scatternet Route Optimization Protocol", in Elsevier: AASRI Procedia, pp 142-148, volume 2, December 2012.

2. "Link Repairing for Inter-piconet Communication Technique in Bluetooth Scatternet", in International Journal of Computer and Electrical Engineering, pp. 583-588, volume 4, August, 2012.

3. Distributed and Hybrid Congestion Handling Protocol for Efficient Bluetooth Network. Journal paper (under review) in Elsevier.

4. An Efficient Route Maintenance Protocol for Dynamic Bluetooth Networks. Journal paper (under review) in Springer.

5. "Bluetooth Network Re-formation Protocol for Reducing Path Length", in proceeding of IEEE: International Conference on Computer and Information Sciences, June 2012.

6. "Least Frequently Used Bridge Reduction Protocol for Bluetooth Scatternet", in proceeding of IEEE. National Postgraduate Conference, Malaysia, 2011.

7. "Route Breakage Prediction Protocol for Bluetooth Network Recovery" in proceeding of IEEE: International Conference on Computing, Information Systems and Communications, Singapore, May, 2012.

8. "Bluetooth Inter-piconet Congestion handling Protocol Through Network Restructuring" in proceeding of Springer: International Conference on Informatics Engineering & Information Science (ICIEIS2011), November, Malaysia.

9. "A Flexible Relay Selection Technique for Bluetooth Scatternet", in proceeding of IEEE: International Conference on Computer and Information Sciences, June 2012.

10. "Self Reorganizing Network in Bluetooth Scatternet" in proceeding of IEEE:

International Conference on Open Systems (ICOS2011), Malaysia, 2011.

11. "Hybrid Congestion handling and route repairing protocol for Bluetooth networks", accepted in APC2013.

APPENDIX A

DEFINITION OF TERMINOLOGIES

Auxiliary Master (AxM) and Auxiliary Bridge (AxB):

An auxiliary master is a slave node in an original piconet but becomes the master node in the new piconet, whereas any intermediate node between the source and destination can be selected as an auxiliary bridge.

Bluetooth Routing devices:

Whenever a Bluetooth device wants to communicate with another Bluetooth device, then first of all, a communication link is established known as a routing link and the intermediate devices which are used to make the routing link are called routing devices. These devices can be slave, master or bridge.

Bluetooth Routing Inter-Piconet:

Whenever a routing link is established within a scatternet, it may pass through multiple piconets; therefore, the routing link consists of several routing master devices. So, a piconet that holds the routing master device is known as the Bluetooth Routing Inter-Piconet.

Device Information Table (DIT):

According to the ERMP, within a scatternet, when a new link is established then the routing master and bridge devices maintain their information tables. The routing master device maintains a Routing Master Information Table (RMIT) that contains the list of slave devices, clock offset, device ID, device status (active/inactive), energy level, signal strength, Fall Back Masters (FBM) and Fall Back Bridges (FBB). Each routing bridge device maintains a Routing Bridge Information Table (RBIT) that contains the list of the connected master devices, device status (active/inactive), signal strength, energy level, list of FBBs, list of FBMs and the degree.

Most Frequently used Nodes (MFN):

During transmission, some devices contain heavy data to share with the other devices therefore they transmit data again and again (frequently) until the complete data exchange.

Node Information Table (NIT):

Each master node maintains a node information table (NIT), which contains the node ID, Clock-offset etc.

Park Mode (PM):

During the role switch operation, the proposed protocol uses the park mode for those nodes which will change their status in new piconets. The purpose of using the park mode is saving active member addresses.

Received Data:

The incoming data is called the received data (Rx), the master device of a piconet keep records each packet it transmits to any device in the piconet.

Threshold:

The threshold value is used for congestion handling on the master and bridge nodes. When MF communicating nodes reach the limit of the threshold value, the proposed protocol performs network restructuring.

Traffic Flow Table (TFT):

The proposed protocol maintains two types of TFT, one for master node and the other for the bridge node called the Master Traffic Flow Table (MTFT) and the Bridge Traffic Flow Table (BTFT). The MTFT keeps the information of all of the incoming and outgoing data traffic going through the master node within the piconet. The BTFT keeps the information of all of the connected masters. This table shows the information of all of the communicating links going through the bridge node.

Transmit Data:

The outgoing data is called the Transmit data (Tx), the master of a piconet records each packet, when any device sends a data packet within or outside the piconet.

Total Data Flow:

The sum of the received and transmitted data is the Total Data Flow (TDF).

Threshold for Signal-to-Noise Ratio (SNR):

The ratio of the signal power to the noise power is known as the Signal-to-Noise Ratio and it is abbreviated as SNR.

Weak Routing Link:

A routing link between two directly connected devices is considered as a weak link if the devices have weak signal strength or they are out of the radio range of each other and cannot communicate properly.

Weak Routing Device:

In the routing path, a device which has the minimum energy level and weak signal strength is considered as a weak routing device, or a device whose energy level is normal but has weak signal strength is also considered as a weak device. It can be a sender or receiver.

APPENDIX B

BASIC PHYSICAL MODEL

In order to understand the configuration and wireless communication procedure of Bluetooth technology for this research work, a basic physical model is developed that validates the basic model. Different kinds of equipment are available for a physical model; Bluetooth Telecomm Trainer is one of the famous physical devices.

BLUETOOTH TELECOMM TRAINER (BTT)

The Bluetooth Trainer (BTT) is a piece of experimental equipment of Bluetooth technology. The step by step experimental procedures are shown in Fig.1. The BTT performs the following functions:

a. Protocol Analysis and Simulation / Emulation.

b. Demo function to simulate the Bluetooth operation.

c. Data communication experiment by using different interfaces, i.e. USB, Universal Asynchronous Receiver/Transmitter (UART).

d. Bluetooth hardware experiment.

e. Bluetooth host programming.

f. Multipurpose usage like Education, Research and Development.

The experiments were conducted on FTP.

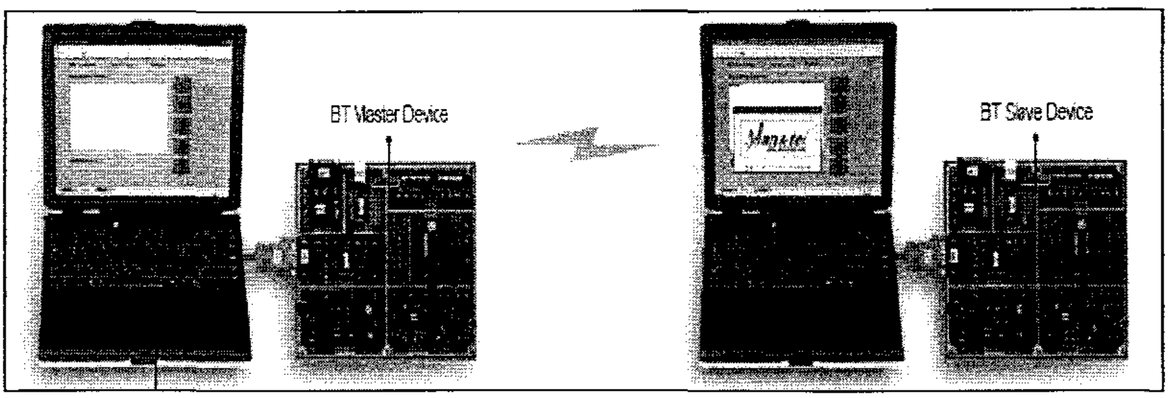


Figure.1: Bluetooth Telecomm Trainer

1. Hardware function

i. RF Module: The Bluetooth RF module is used for the communication of Bluetooth mobile devices. This module consists of a BlueCore02 chip, flash ROM, external memory, crystal oscillator, a Low Noise Amplifier (LNA) and an antenna.

ii. HCI interface: The transmission between the HCI controller and the HCI host is attained by the UART or USB interface (Kazemian and Meng 2006).

a) UART: A Universal Asynchronous Receiver/Transmitter is a sort of asynchronous receiver/transmitter and a part of a computer hardware that interprets data between parallel and serial structures. With the help of UART, Bluetooth technology gives an interface for communication with the HCI controller of a Bluetooth module. It can connect a PC with the UART serial cable. Rx, Tx, RTS, and CTS are marked *n*, the basis of the Bluetooth RF unit standard; the Rx specifies from the PC to the RF unit data and the Tx from the RF unit to the PC data conversely. Furthermore, RTS and CTS can be used for conversation between Bluetooth devices.

b) USB: The BT technology presents a USB interface to hold the HCI controller and the high speed Bluetooth RF module. The USB interface can also be used for the power supply to the terminals.

c) SPI connection: The SPI interface can be used to adjust the Persistent Store (PS) value of the flash memory. The SPI interface can make a link with the CON3.

2. Software

BTT is used to observe packet analysis programming and emulation programming for Bluetooth protocols like RFCOMM, SDP, L2CAP, and HCI. The VC++ programming covers the operations of Reset, Inquiry, HCI, Connection etc.

2.1. Analysis Mode

The analysis mode provides the general progress of the Bluetooth protocols. At each layer, keys are connected in a special order to activate the Bluetooth module.

a. The Bluetooth device's role is checked by pressing the Master Role or Slave Role key in the analysis Tab.

b. It is necessary to click the HCI CON key of that device in which a Master Role has been activated. After the Master Role selection the Master Indication System requests information and a device having the Slave Role Indication System receives the communication. If there are multiple devices then one of them is selected as a slave device.

c. The reset key is used to switch the role of devices from Master Role to Slave Role or Slave Role to Master Role.

The packet structure is monitored by the analysis mode; Fig.2 shows that this mode is specially structured to study the details of the sequence, conditions and organization of the Bluetooth protocol layers.

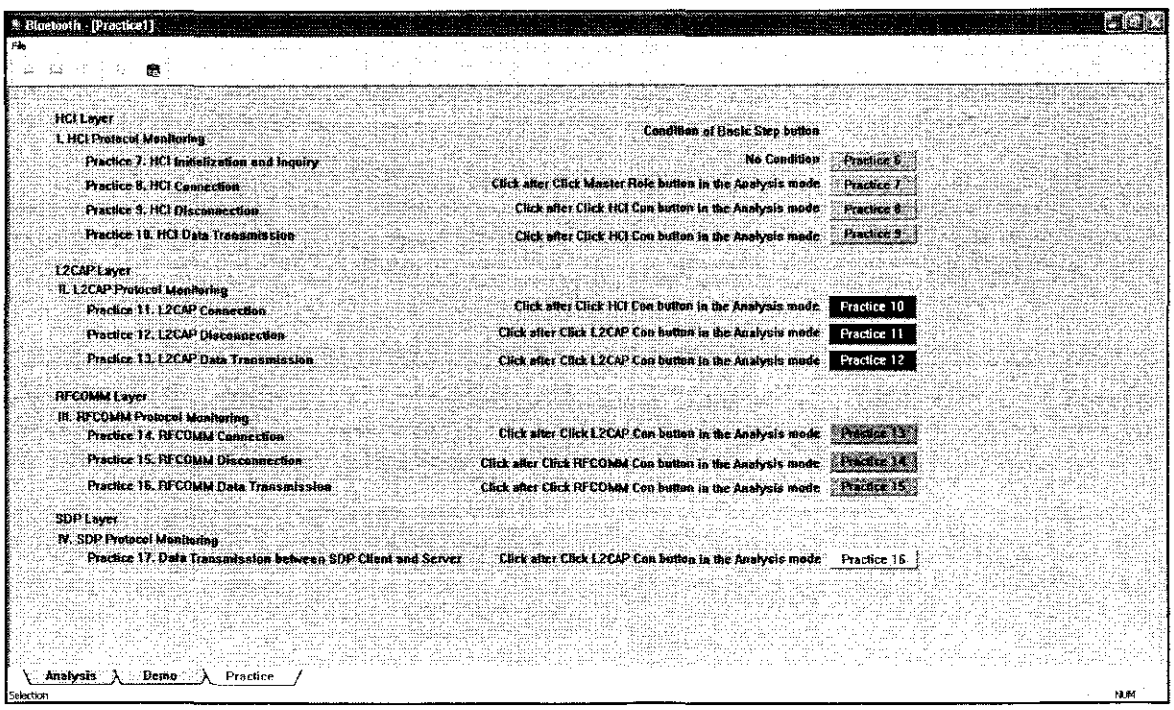


Figure.2: Protocol layers

3. Usage of Program

We used to connect the 9-pin female connector of the UART cable to the COM1 port of the PC and connect the other 9-pin male connector to the UART connector of the BT. In the next step we opened a protocol window; choose the H4 from the protocol box, as shown in Fig.3.

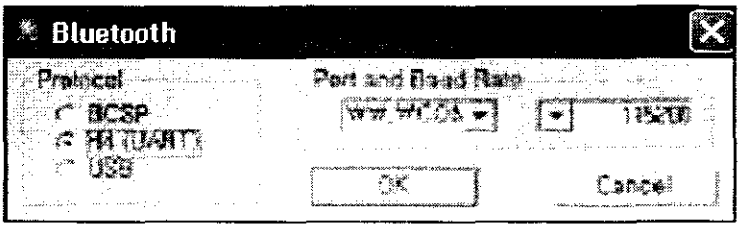


Figure.3: Protocol window

After that we selected RF Test Mode and select the RADIO STATUS to execute as shown in Fig.4.

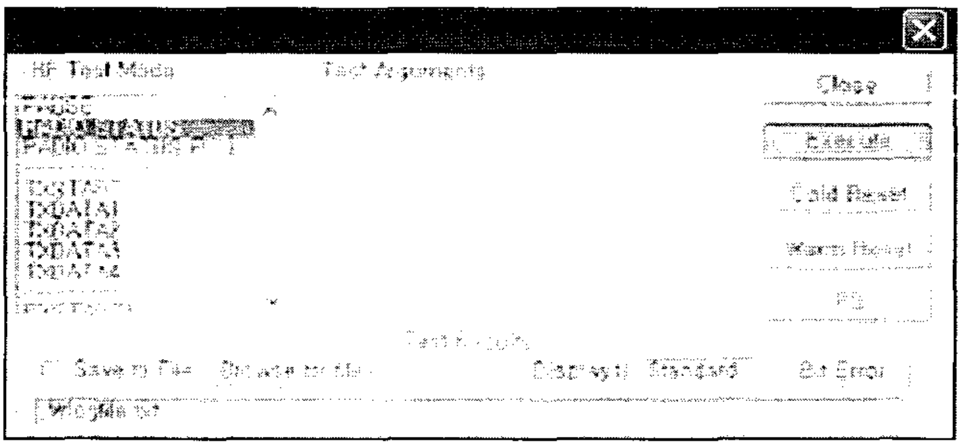


Figure.4: Bluetooth RF test

4. Bluetooth Telecommunication Setup

This step is concerned with the commands, event execution process and analysis of the Bluetooth systems. How to execute the protocol monitoring program by transmitting commands and receiving any event?

a. Data transmission

In the first step, the command is passed from a PC to the BT and the event is transmitted from the BT to the PC. Now, ACL and SCO data packets are exchanged between the BT and the PC. The reset command is used to initialize the Bluetooth Host controller, Link Manager, and the radio unit as the software is running on the PC and will not send out any data with RF. As a response, the Bluetooth device transmits a Command Complete Event to the PC.

b. Inquiry of the Surrounding Bluetooth system

As the Bluetooth devices always communicate wirelessly, therefore a protocol is needed that discover the devices which are in the proximity of other devices. Bluetooth devices use a technique called inquiry for discovering other devices within the radio range of 10m (Jung et al. 2007). An inquiry command is used to enter the Bluetooth devices in the inquiry mode that helps to discover the other Bluetooth devices which are in the proximity of each other (Wang et al. 2009).

c. Communication between PC and BT

The PC and the BT communicated through the UART method. Packets are sending or receive through these methods.

d. Command for slave role

First of all, a Bluetooth device has to verify its own Bluetooth Device Address (BD_Addr) and then executed the inquiry procedure. All commands used for the communication of HCI hold a packet header and hexa code (0xo01 in the UART communication).

e. Ready for Paging Response

The slave device is set for getting the response from the master device for paging mode (Chakrabarti et al. 2004). In this step, device roles (master or slave) are apparently selected to the Bluetooth devices. If it starts without selecting a master/slave role, then this command will be implemented on both Bluetooth systems. The device which is given the role of the master should also execute the Inquiry Command.

f. Ready for Inquiry Response

The command to allow the Bluetooth device is set for an Inquiry Response must write the Inquiry Scan command. This command is like the parameter setup of the paging response. The Inquiry Response time shows how long it takes for an Inquiry Response and the total Inquiry Response standby time.

g. Command for Master Role

A master device finds the Local BD_Addr. Master device send out a signal to access the free space from slave buffer. When a BT slave device receives this signal, it transmits its available buffer space back to the Host. In this way, it stops the host for transmitting more data so that the data loss from the Buffer Overflow can be blocked.

5. Bluetooth Telecommunication Experiments

The analytical execution of experiments consists of the following three steps:

Step 1: To understand the HCI Protocol Data Unit (PDU), hold the flow by entering the data.

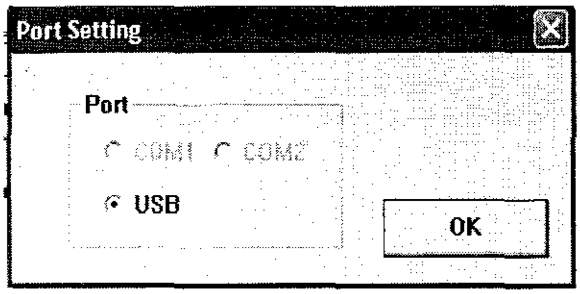
Step 2: Keep the HCI command and event packets in a buffer and arrange an absolute protocol packet form.

Step 3: Write a program to observe the process in which the Bluetooth system can communicate successfully.

i. Step 1 Experiment

In this step, a command is entered and checked the output events. The command value contains the Op-code with some parameters. To initiate a command with different parameters, the following procedure is executed that refers to the above detailed description.

a) We Run the Bluetooth and selected the monitor key. The COM port is at this time connected to the UART cable and now clicked on the OK button, as shown in Fig.5.



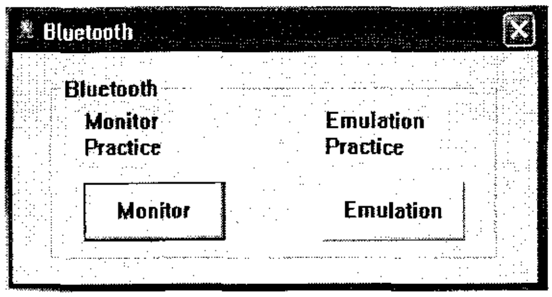


Figure.5: Mode selection and port

b) In the next step we opened the practice tab and clicked on the Practice 6 button. The following screen is displayed as shown in Fig.6.

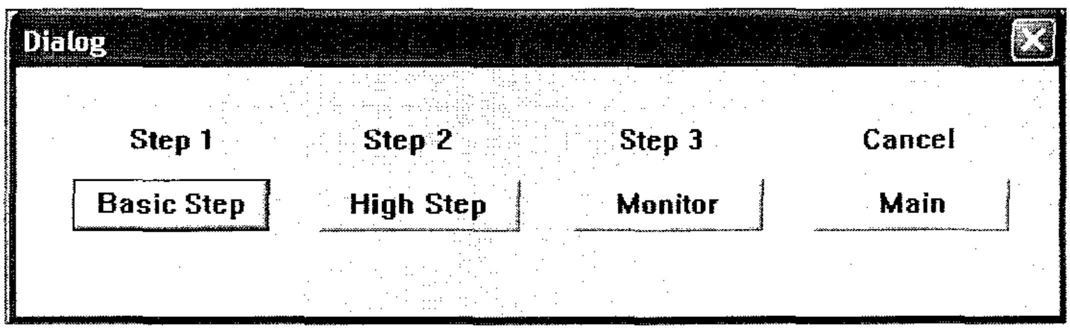


Figure.6: Dialog selection

c) Selected Step 1, the following screen is opened as shown in Fig.7.

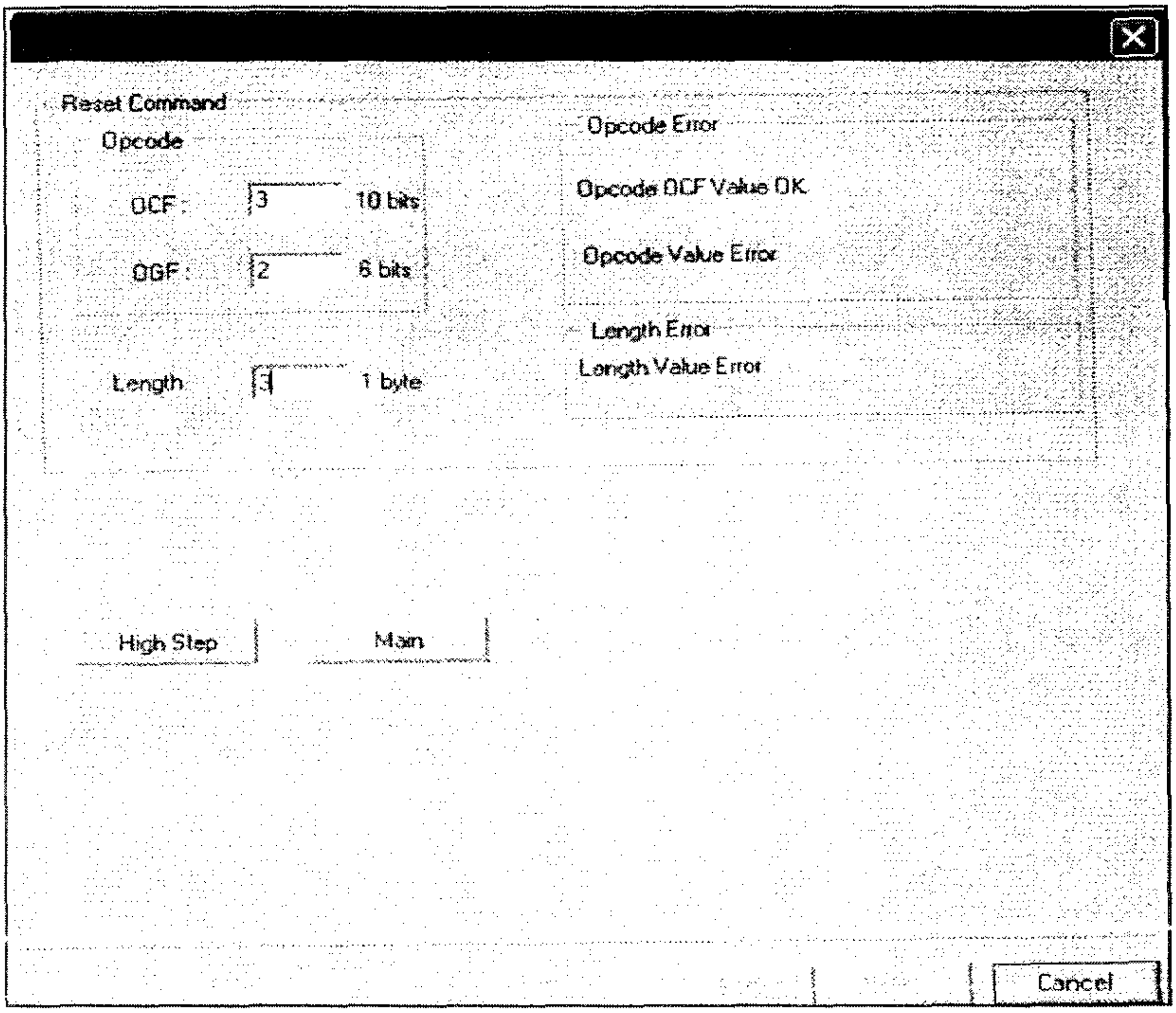


Figure.7: Reset command selection

d) The decimal number is entered as paramcter value. We ticked the Next button, the reset command is passing from the PC to the BT and an IAC command window is opened. This command generates the same results as clicking the reset button, as shown in Fig.8.

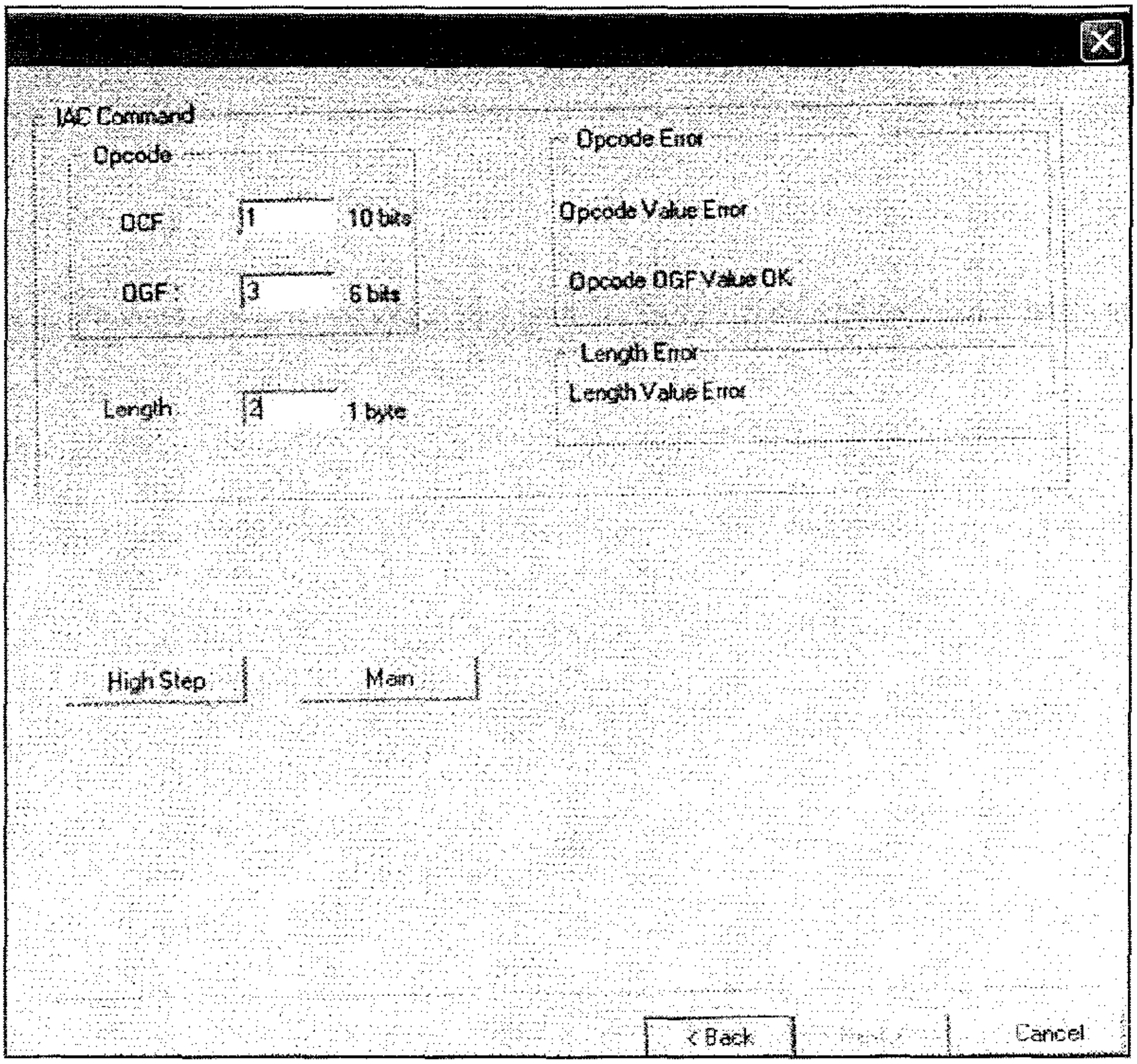


Figure.8: IAC command selection

e) An Inquiry command window is opened after entering the correct values. As all required values are entered and pressed the Finish key. Then the BT shows the same results as the Master role. Analysis tab is used to check the event response.

ii. Step 2 Experiment

HCI creates a complete understanding of the command and the event packet parameters. We selected the practice 6 button in the practice tab and choose step 2, as shown in Fig.9.

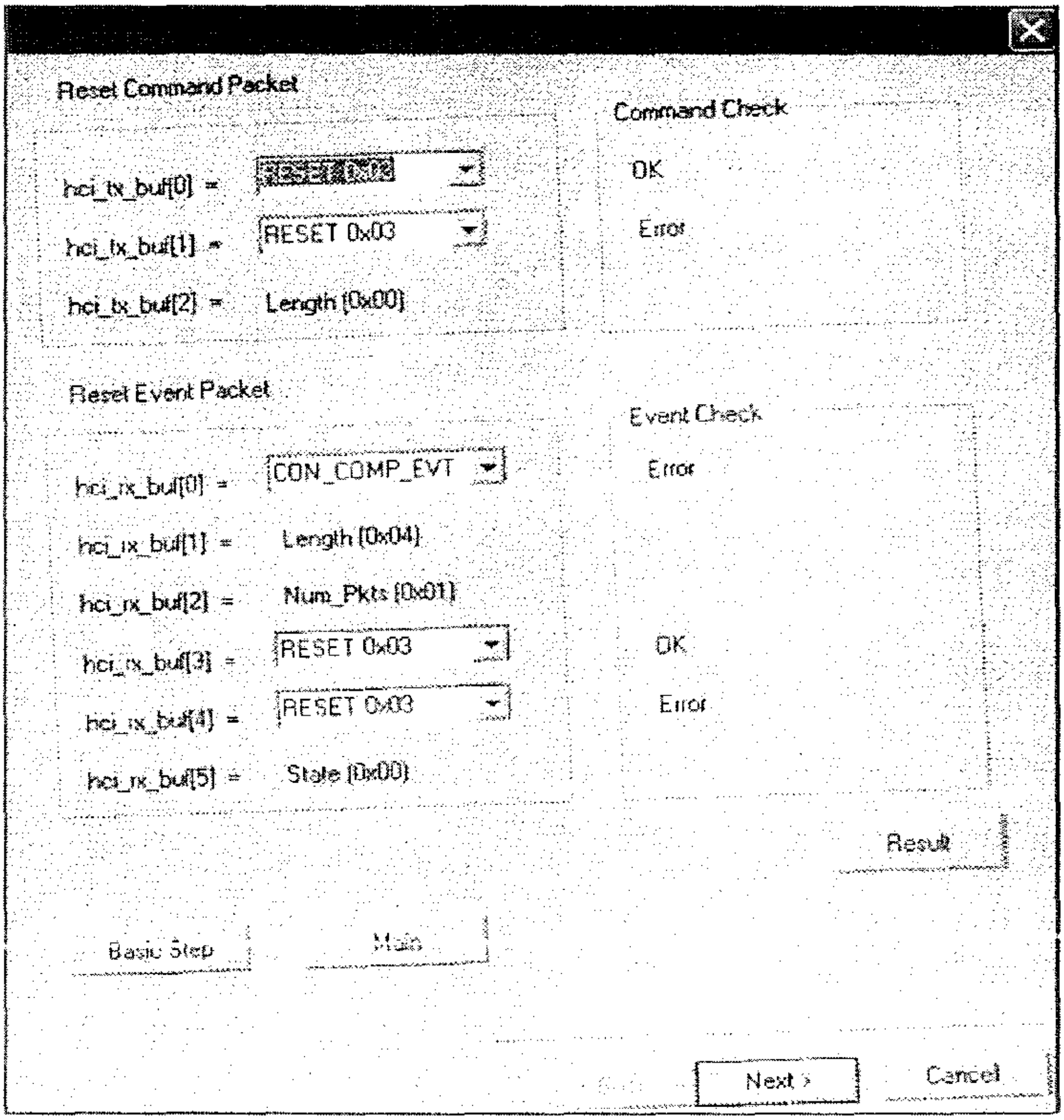


Figure.9: HCI Reset

We selected an item, and then clicked the result button. In the present event window, the practice is executed in order and then clicked the Finish button.

iii. Step 3 Experiment

The objectives of step1 and step 2 are used to realize the Bluetooth protocol data and the commands are used to check their results, whereas step 3 generates a monitoring program for the flow of the data processing. Many routines are used to swap the HCI commands, HCI events and HCI data packets for the communication of a Bluetooth practical program between the PC and the BT. When click on the Master Role Select key, the subroutines of the AnalysisTxD11 and AnalysisRD11 are called on. At the same time, the AnalysisTxD11 is called on by sending a command from the host to the host controller (PC to BT). The AnalysisRxD11 is called on by sending an event from the BT to the PC. During this process, protocol observation is planned based on the following parameters.

a) Run the Bluetooth; choose the Monitor key and the COM port.

b) Choose the practice 6 key from the practical tab and click the Step 3 monitor. It executed programming cpp automatically, as shown in Fig.10. On the first time it runs the program Bluetooth_EDU.exe.

AnalysisTxD11 is the command or the data sent out through the UART cable from the PC to the BT, and the AnalysisRxD11 is the Event of the data received from the BT to the PC.

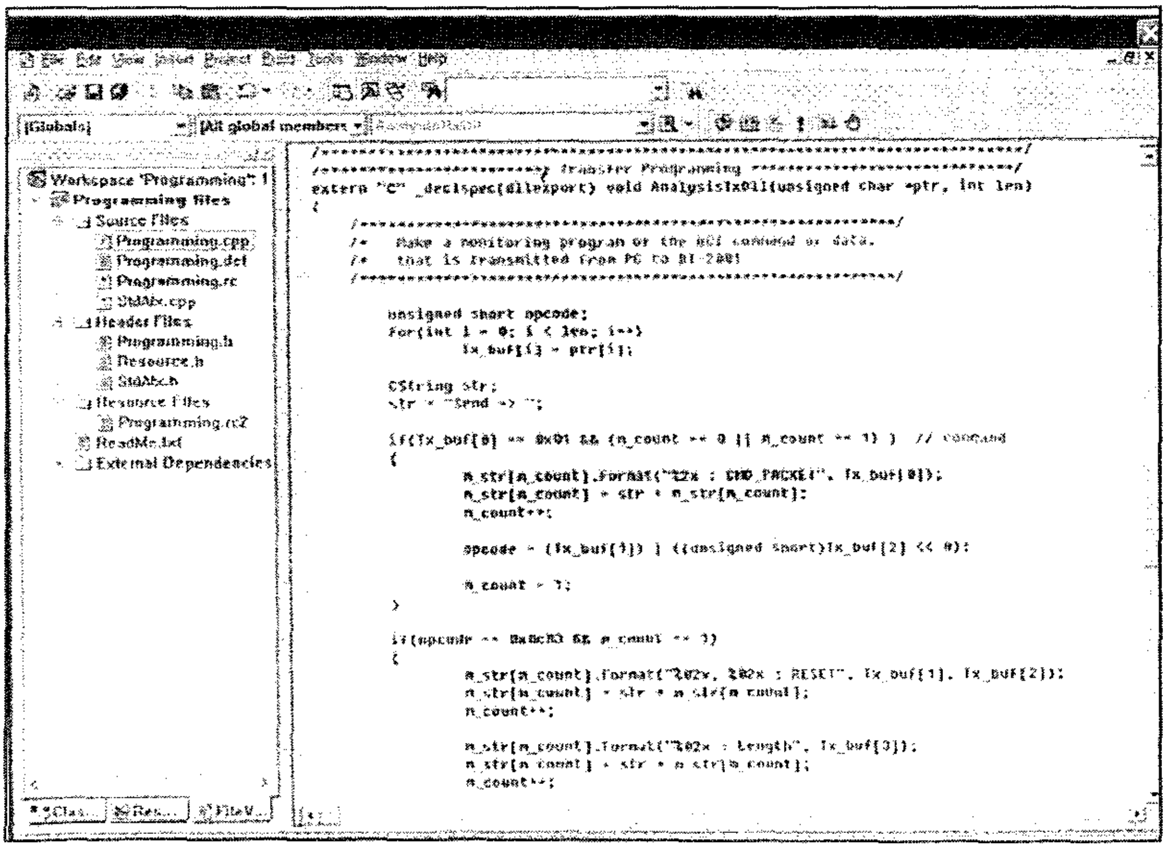


Figure.10: Programming window

We compiled and executed the program, and selected the Reset button in the edit menu of the Analysis tab. The Reset command and the event results are monitored on the window. Finally, we record the monitored results in the Report.

6. Physical model performance study

This section presents a complete quantitative evaluation of the basic physical model and basic simulation model. In order to setup the physical network Bluetooth Telecommunication (BT) devices were used, and for the simulation, UCBT was used. UCBT was based on an open source simulator NS-2 that provided an absolute simulation model for the Bluetooth performance assessment on Bluetooth layers. UCBT directly approximates Bluetooth protocols; it is an open source that can be utilized for the implementation of new ideas. UCBT provided a simulation capacity on different layers of the baseband and above, like LMP, BNEP and L2CAP. The Bluetooth functionality was defined in the BT package, which used the basic functionality of the Bluetooth. Output files were generated after the successful execution of the experiments. A total four fixed devices were used. The energy consumption for the transmit and receive was assumed to be 0.0763 x 10-6 J/bit (Cui et al. 2010). The source and destination were selected; each device could participate in multiple connections. The experiments were implemented five times and the results were taken by averaging of those five times.

7. The validation of Basic Simulation model

To verify the basic simulation model, the results of the basic simulation model and the results of the physical model are compared as below. Four physical Bluetooth devices are used to calculate the throughput, energy consumption and delay. Data are transmitted from one device to the last device. An average 1 GB data are transmitted from the first device to the second device, once date are received successfully the same data are transmitted to the next node. Simulation data are also transmitted from the source node to the destination.

7.1. Average throughput

The throughput was considered as the successful rate for the delivered packets across the network or the number of successful data bytes received by a node per unit time. Average 1GB data are transmitted from one Bluetooth device to the last Bluetooth device. It was analyzed total transmission time from one Bluetooth device to next Bluetooth device was 20min. Total transmitted data are converted to kilobytes and divided by total time from one hop to next hop. It was observed that when the number of hops increased, it decreased the throughput in the network as shown in Fig.11. The reason is that each device required some synchronization time and en-queued and de-queued the received packets. From the comparison of both results, it was analyzed that the throughput of the simulation and the physical model were almost same.

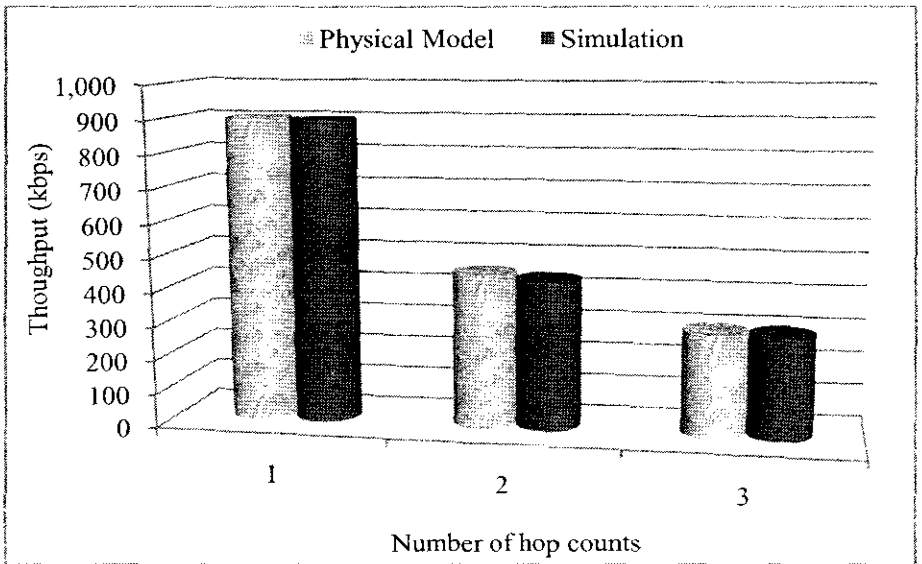


Figure.11: Number of hop counts vs. Average throughput

7.2. Control overhead

Control message overhead is considered as the sum of (MAC + bb + POLL + NULL) packets for the received data packets. It is analyzed that an average each hop used 500 control overhead packets, as an average each packed size was 12Kb. It was observed that when the number of hops increased it also increased the control packets as shown in Fig.12. Whenever a new route request came or a new device activated it increased the control overhead. From the comparison results, it was analyzed that the physical model had a bit higher control overhead, due to the retransmission of missing packets.

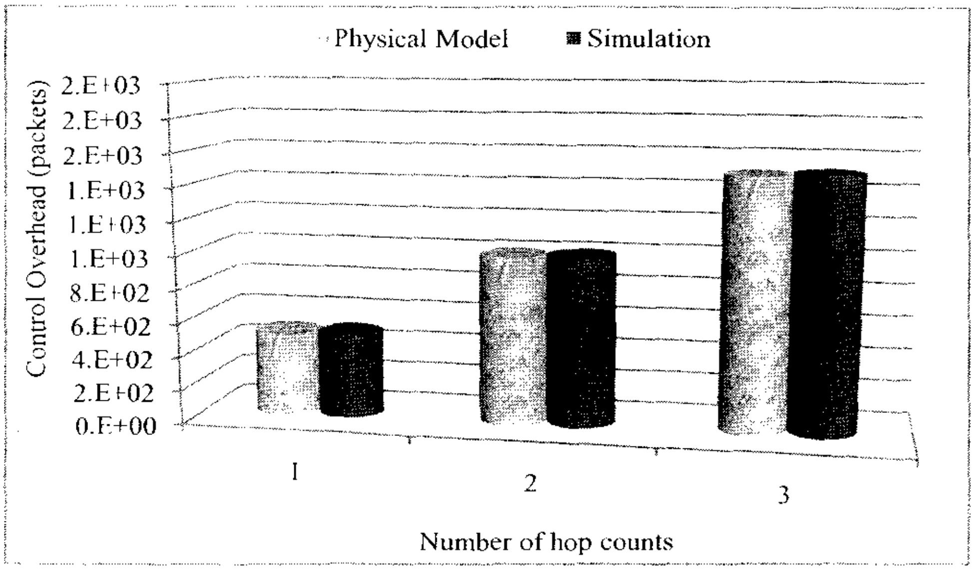


Figure.12: Number of hop counts vs. Control overhead

7.3. Total Delay

The network delay indicates how much time is required for transmitting a bit from a source device to a destination device. The total end-to-end delay is the sum of (packet received time – packet transmitted time). Other aspects may have an effect on the delay, like queue length, executing time, transmission time and distribution time. Since the number of hops increased, the delay time increased as well. During data transmission between physical devices, it is analyzed that total delay per hop was 1200s. From Fig.13, it was analyzed that the delay of the physical model and the simulation was almost the same for a different number of hops.

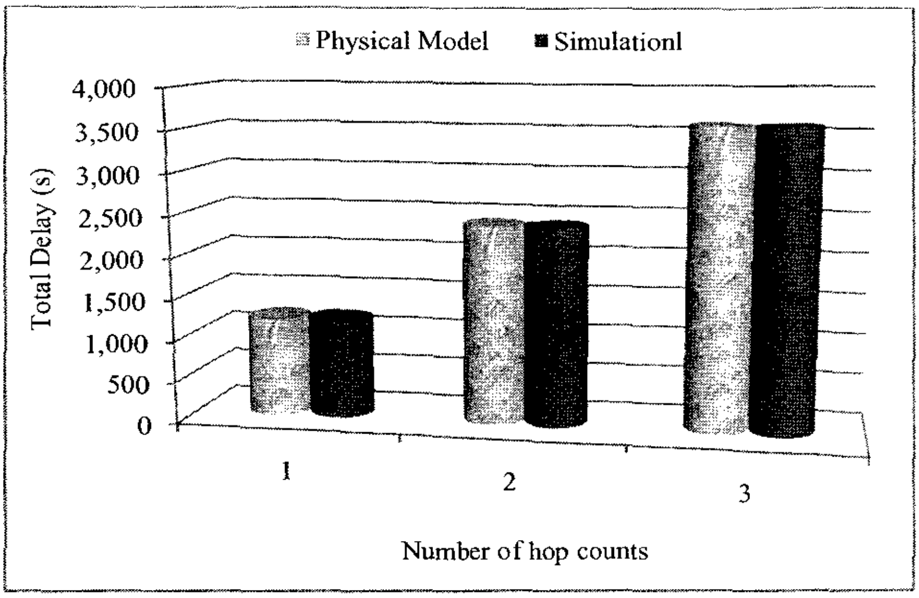


Figure.13: Number of hop counts vs. Delay

8. Summary

The validation of basic physical model is presented in this section. In general, as a longer route utilizes the maximum resources, it create maximum amount of control overhead in the network. On the other side, a short route utilizes the system resources efficiently. For Bluetooth communication, the physical model and the basic simulation performances were compared. Through the comparison of the basic simulation model and physical model, it was observed that both of the methods had almost the same results of delay, message overhead, energy consumption and throughput.