ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

COOPERATIVE COMMUNICATION IN MOBILE ADHOC NETWORKS USING DYNAMIC SOURCE ROUTING PROTOCOL

By: Mosisa Ebsa Barento
Advisor: Dr. Murad Ridwan

Thesis submitted to the School of Electrical and Computer Engineering

This Thesis is submitted in Partial Fulfilment of the Degree of Master of Science in Electrical and Computer Engineering
(Communication Engineering Stream)

Addis Ababa, Ethiopia
May 2016
Addis Ababa University

ABSTRACT

A mobile Ad-hoc network (MANET) is consists of mobile platforms which are simply referred to as "nodes" which are free to move about arbitrarily without requiring centralized administration or fixed network infrastructure. The nodes may be in or on airplanes, ships, trucks, cars, perhaps even on people or very small devices, and there may be multiple hosts per router. Through cooperatively forwarding packets for each other, nodes in mobile Ad-hoc networks can communicate with other nodes out of their direct transmission ranges. MANET nodes are equipped with wireless transmitters and receivers using antennas which may be omnidirectional (broadcast), highly-directional (point-to-point), possibly steerable, or some combination thereof. In many situations, such as military or emergency applications, nodes in an Ad-hoc network belong to the same authority and pursue the common goals. Under such circumstances, fully cooperative behavior, such as unconditionally forwarding packets for each other, can be assumed. We refer to such Ad-hoc networks as cooperative Ad-hoc networks.

The common theme of most research in this area is to optimize physical layer performance measures without considering in much detail how cooperation interacts with higher layers and improves network performance measures. Because these issues are important for enabling cooperative communications to practice in real-world networks, especially for the increasingly important class of mobile Ad-hoc networks.

Cooperative routing technology can efficiently improve the performance of the network by initiatively selecting cooperative nodes to forward the data. *In this research, cooperative routing algorithms is proposed and implemented in mobile Ad-hoc networks and the performance of the proposed cooperative Ad-hoc algorithms are compared with non-cooperative Ad-hoc dynamic-source-routing (DSR) protocol using the network simulator (NS-2).*

Keywords: *Cooperative Communication, Mobile Ad-hoc Network, Dynamic Source Routing Protocol, MANET, DSR, NS-2.*

School of Graduate Studies

This is to certify that the thesis prepared by Mosisa Ebsa, entitled: *Cooperative Communication in Mobile Ad-hoc Networks Using Dynamic Source Routing Protocol* and submitted in partial fulfillment of the requirements for the *Degree of Masters in Science (Communication Engineering)* complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examination Committee:

Advisor _____ Signature _____

Date _____

Examiner _____ Signature _____

Date _____

Examiner _____ Signature _____

Date _____

_____

Chair of Department of Graduate Program Coordinator

Declaration

I the undersigned, declare that this MSc thesis is my original work, has not been presented for fulfillment of a degree in this or any other university, and all sources and materials used for the thesis have been acknowledged.

Name: Mosisa Ebsa

Signature: _____

Place: Addis Ababa University

Date of submission: _____

This thesis work has been submitted for examination with my approval as a university advisor.

Murad Ridwan (PhD) _____

Advisor's Name Signature

TABLE OF CONTENTS

ABSTRACT i
ACKNOWLEDGMENT vi
LIST OF ACRONYMS vii
LIST OF FIGURES viii
CHAPTER ONE 1
1. OVER VIEW OF MOBILE AD-HOC NETWORK 1
1.1 INTRODUCTION 1
1.2 BACKGROUND OF MOBILE AD-HOC NETWORK 2
1.3 WHAT IS MOBILE AD-HOC NETWORK? 3
1.4 MOBILE AD-HOC NETWORKING TECHNOLOGIES 4
1.5 MOBILE AD-HOC NETWORK APPLICATIONS 8
1.6 TOPOLOGY MANAGEMENT IN MOBILE AD-HOC NETWORKS 9
1.7 THE OSI LAYER IN MOBILE AD-HOC NETWORK 10
1.8 SELECTING BETWEEN INFRASTRUCTURE AND AD-HOC MODES 14
1.9 MOTIVATION FOR COOPERATIVE RETRANSMISSION 15
1.10 COOPERATIVE TRANSMISSION 16
CHAPTER TWO 17
2. ROUTING IN MOBILE AD-HOC NETWORK 17
2.1 PROACTIVE ROUTING PROTOCOLS 17
2.2 REACTIVE ROUTING PROTOCOLS 19
CHAPTER THREE 25
3. DATA LINK LAYER 25
3.1 MEDIUM ACCESS CONTROL 25
3.2 IEEE 802.11 DCF SCHEME DESCRIPTION 30
3.3 COOPERATIVE COMMUNICATION IN MAC LAYER 31
CHAPTER FOUR 36
4. DESCRIPTION AND RELEVANCE OF DSR PROTOCOL 36
4.1 OVERVIEW AND IMPORTANT PROPERTIES OF THE PROTOCOL 37
4.2 COOPERATIVE IMPLEMENTATION IN DSR PROTOCOL 43
CHAPTER FIVE 45
5. RESULTS 45
5.1 BACKGROUND ON NS-2 SIMULATOR 45
5.2 PARTS NEEDED BY ONE SIMULATION IN NS-2 47
5.3 SIMULATION ANALYSIS OF COOPERATIVE RETRANSMISSION 48

CHAPTER SIX 59
6. CONCLUSION AND RECOMMENDATION 59
6.1 CONCLUSION 59
6.2 RECOMMENDATION AND FUTURE WORK 60
APPENDIX 61
REFERENCES 63

ACKNOWLEDGMENT

I would like to thank my advisor Dr. Murad Ridwan for his advising, cooperation and fellowship, and for dedicating his time whenever I want to meet him and needs his help. Above all, for encouraging and pushing me to finish this long journey work of mine. I also want to extend my thanks to Dr. Yalemzewd Negash for supporting and directing me at the beginning of this work.

And finally, I want to thank my father and mother for been compassionate to me and helping me in all aspects they could at their best effort. My beloved wife Destu Yassin is at the heart of my success by being with me physically and expressively, hence my gratitude is beyond announcement.

LIST OF ACRONYMS

ACK	Acknowledgment
AODV	Ad-hoc on-Demand Distance Vector
ALOHA	Areal Locations of Hazardous Atmospheres
ABR	Associativity-Based Routing
CSMA	Carrier Sense Medium Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CFC	Clear For Cooperation
CTS	Clear To Send
CGSR	Cluster Head Gateway Switch Routing
CBR	Constant Bit Rate
CW	Contention Window
Coop-Adhoc	Cooperative Ad-hoc
DSDV	Destination-Sequenced Distance Vector Routing
DSSS	Direct Sequence Spread Spectrum
DCF	Distributed Coordination Function
DIFS	Distributed Coordination Function Inter Frame Space
DSR	Dynamic Source Routing
ISM	Industrial Scientific and Medical
IFS	Inter-Frame Space
IEEE	International Electrical and Electronic Engineering
LCC	Least Cluster Change
LAN	Local Area Network
LLC	Logical Link Control
MACSDU	Mac Service Data Unit
MAC	Medium Access Control
MRL	Message Retransmission List
MANET	Mobile Ad-hoc Network
NS-2	Network Simulator Version 2
NRL	Normalized Routing Load
OSI	Open Systems Interconnection
OFDM	Orthogonal Frequency Division Multiplexing
PDF	Packet Delivery Fraction
PRNET	Packet Radio Network
PAN	Personal Area Network
PC	Personal Computer
PHY	Physical Layer
PCF	Point Coordination Function
PIFS	Point Coordination Function Inter Frame Space
RFC	Request for Cooperation
RTS	Request To Send
SIFS	Short Inter Frame Space
SURAN	Survivable Adaptive Radio Networks
TORA	Temporally-Ordered Routing Algorithm
TDMA	Time Division Multiple Access
WLAN	Wireless Local Area Network
WPAN	Wireless Personal Area Network
WRP	Wireless Routing Protocol

LIST OF FIGURES

Figure 1: An Infrastructure Mobile Network [2] 1
Figure 2: Mobile Ad-hoc Network [2] 2
Figure 3: An Ad-hoc Network Example among Laptops [2] 4
Figure 4: Mobile Ad-hoc Network Topology Examples [7] 9
Figure 5: OSI Layer in Mobile Ad-hoc Network [9] 10
Figure 6: Routing Protocols in Mobile Ad-hoc Network [15] 17
Figure 7: Route Record during Route Discovery [15] 20
Figure 8: Propagation of Route Reply with Route Record [15] 21
Figure 9: MAC Layer in IEEE 802.11 Standard [26] 27
Figure 10: IEEE 802.11 MAC timing (Basic access method) [26, 27] 29
Figure 11: IEEE 802.11 DCF basic access scheme [27] 30
Figure 12: IEEE 802.11 DCF RTS/CTS scheme [27] 31
Figure 13: Cooperative communication in RTS/CTS scheme 34
Figure 14: Flow chart for DSR routing mechanism [28] 38
Figure 15: Route Discovery Example: **A** is the initiator and **E** is the target [29] 39
Figure 16: Flow chart for Route Request Mechanism [28] 40
Figure 17: Flow chart for Route Reply Mechanism [28] 41
Figure 18: Route Maintenance: Node **C** is unable to forward a packet to next hope node **D** [29] 43
Figure 19: Overall architecture of NS-2 [32] 46
Figure 20: Data flow of one time simulation in NS-2 [33] 46
Figure 21: Layered architecture of NS-2 [33] 47
Figure 22: NS-2.35 interface 49
Figure 23: Throughput Vs Number of Nodes 51
Figure 24: PDF Vs Number of Node 51
Figure 25: NRL Vs Number of Node 52
Figure 26: E-E-Delay Vs Number of Node 52
Figure 27: Throughput Vs Pause Time 53
Figure 28: PDF Vs Pause Time 54
Figure 29: NRL Vs Pause Time 54
Figure 30: End to End Delay Vs Pause Time 55
Figure 31: Percentage Change Comparison 58

CHAPTER ONE

1. OVER VIEW OF MOBILE AD-HOC NETWORK

1.1 INTRODUCTION

Since their emergence in the 1970s, wireless networks have become increasingly popular in the computing industry. This is particularly true within the past decades which has seen wireless networks being adapted to enable mobility. There are currently two variations of mobile wireless networks which are defined by IEEE 802.11 standards. The first is known as infrastructure networks, i.e., those networks with fixed and wired gateways. The bridges for these networks are known as base stations. A mobile unit within these networks connects to, and communicates with, the nearest base station that is within its communication radius. As the mobile travels, out of range of one base station and into the range of another, a handoff occurs from the old base station to the new, and the mobile can continue communication seamlessly throughout the network. Typical applications of this type of network include wireless local area networks (WLANs) [1].

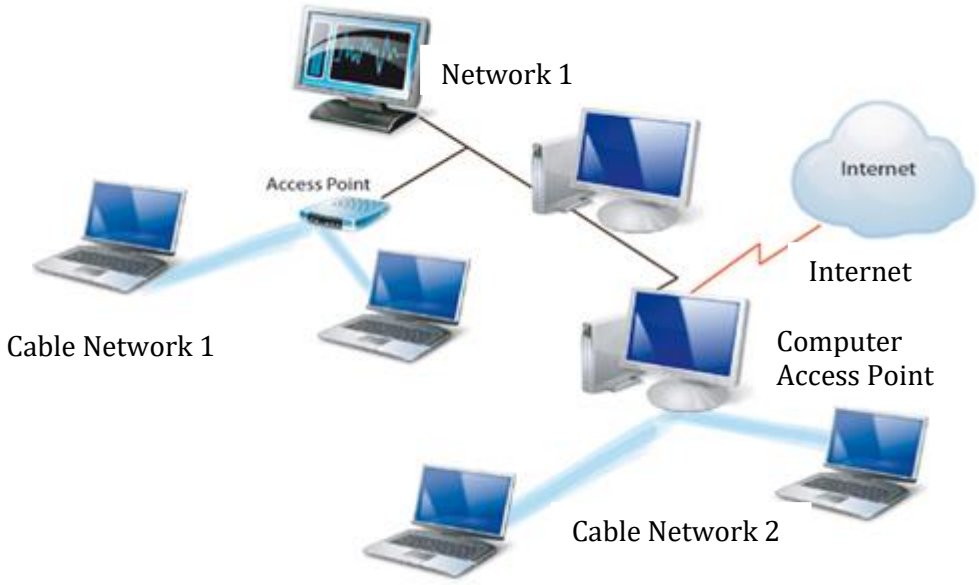


Figure 1: An Infrastructure Mobile Network [2]

The second type of mobile wireless network is the infrastructure-less mobile network, commonly known as Ad-hoc network or Mobile Ad-hoc Network (MANET). Infrastructure-less networks have no fixed routers; all nodes are capable of movement and can be connected dynamically in an arbitrary manner. Nodes of these networks function as routers which discover and maintain routes to other nodes in the network.

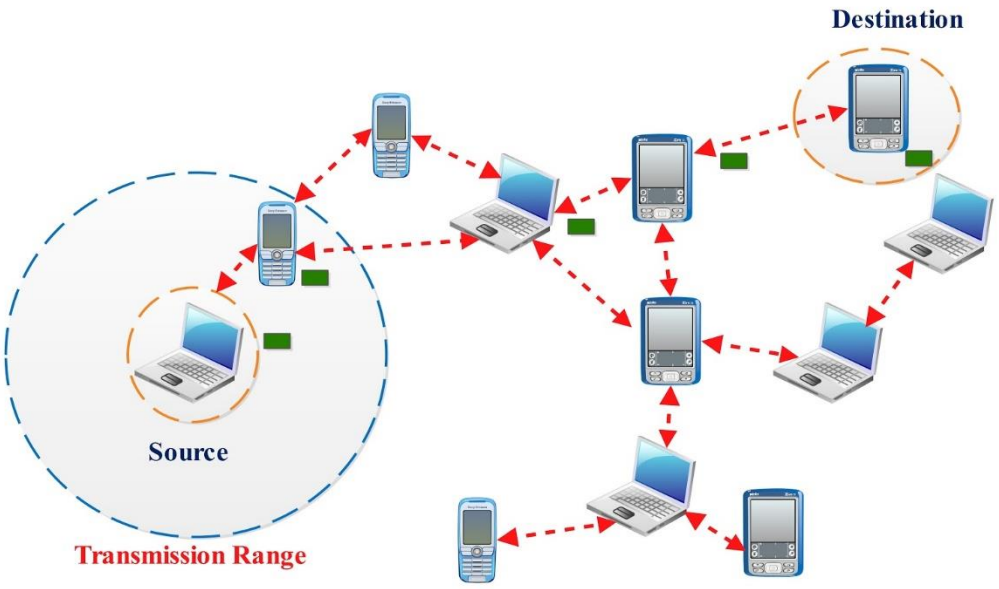


Figure 2: Mobile Ad-hoc Network [2]

This thesis is mainly concerned with cooperative route failure maintenance in the infrastructure-less mobile network i.e. Ad-hoc network, using the Dynamic Source Routing Protocol. It attempts to find an alternative route maintenance mechanism for better performance of data exchange among communicating nodes.

1.2 BACKGROUND OF MOBILE AD-HOC NETWORK

The whole life-cycle of mobile Ad-hoc networks could be categorized into the first, second, and the third-generation Ad-hoc networks systems. Present Ad-hoc networks systems are considered as the third generation. The first generation goes back to 1972. At that time, they were called PRNET (Packet Radio Networks) research for military purpose in 1970s, which evolved into the Survivable Adaptive Radio Networks (SURAN) program in the early 1980s. In conjunction with ALOHA (Areal Locations of Hazardous Atmospheres) and CSMA (Carrier Sense Medium Access), approaches for medium access control and a kind of distance-vector routing PRNET, were used on a trial basis to provide different networking capabilities in a combat environment, [1, 3].

The second generation of Ad-hoc networks emerged in 1980s, when the Ad-hoc network systems were further enhanced and implemented as a part of the SURAN program. This provided a packet-switched network to the mobile battlefield in an environment without

infrastructure. This program proved to be beneficial in improving the radios' performance by making them smaller, cheaper, and resilient to electronic attacks, [1].

In the 1990s, the concept of commercial Ad-hoc networks arrived with notebook computers and other viable communications equipment. The idea of mobile Ad-hoc networks has been under development from 1970s and 1980s in the framework of Mobile Packet Radio Technology (PRNET-1973) and Survivable Adaptive Networks (SURAN-1983). In the middle of 1990's, with the definition of standards, commercial radio technologies have begun to appear and the wireless research community identified in Ad-hoc networks a challenging evolution of wireless networks [1, 4].

Today's emerging standards and technologies for constructing a mobile Ad-hoc network are IEEE 802.11, Bluetooth, and ETSI Hiperlan/2. The deployment of mobile Ad-hoc networks opens a wide-range of potential utilization from military to miscellaneous commercial, private and industrial scenarios, [4].

1.3 WHAT IS MOBILE AD-HOC NETWORK?

Per the IEEE802.11 specification, a mobile Ad-hoc network is a network composed solely of nodes within mutual communication range of each other via the wireless medium. This definition implies that an Ad-hoc network is a complete one, in the sense that all nodes are in the neighbor of each other to form an arbitrary topology without a fixed infrastructure. The nodes move randomly and organize themselves arbitrarily in such a way that the wireless network topology may be subject to frequent changes. Such a network may operate in a standalone fashion, or may be connected to the larger Internet. The nature of Ad-hoc networks makes it suitable for emergency situations such as natural or human-induced disasters, military operations or emergency medical conditions, community networking and interaction among meeting attendees or students during a lecture.

In a multi hop wireless Ad-hoc network, mobile nodes cooperate to form a network without using any infrastructure such as access points or base stations. Instead, the mobile nodes forward packets for each other, allowing communication among nodes outside wireless transmission range. The nodes' mobility and the fundamentally limited capacity of the wireless medium, together with wireless transmission effects such as attenuation, multipath propagation, and interference, combine to create significant challenges on Ad-hoc routing protocols, [4].

Figure 3: An Ad-hoc Network Example among Laptops [2]

1.4 MOBILE AD-HOC NETWORKING TECHNOLOGIES

There exist several technologies which enable Ad-hoc networking. The most popular one is IEEE802.11, used in Ad-hoc mode. IEEE802.11 is commonly referred to as "Wi-Fi". Protocols such as Hiperlan, ZigBee etc., provide alternatives solutions to mobile Ad-hoc networking. A common feature of most wireless networking technologies is that they operate in the unlicensed Industrial Scientific and Medical (ISM) 2.4GHz band. Because of this choice of frequency band, the network can suffer interferences from microwave ovens, cordless telephones, and other appliances using this same band plus, of course, other networks. Mobile Ad-hoc networking uses the following technologies [3].

1.4.1 Packet Radio

Packet radio was used for the earliest versions of mobile Ad-hoc networks. It was sponsored by DARPA in the 1970s. It allows the transmission of digital data over radio channels. Using special radio equipment, packet radio networks allowing transmissions at 19.2 Kbps, 56 Kbps, and even 1.2 Mbps have been developed [3].

1.4.2 IEEE802.11

Wi-Fi is a wireless networking technology based on the IEEE802.11 specifications. The first and still most used Wi-Fi standard is referred to as IEEE802.11b. It was then distributed into IEEE802.11a, IEEE802.11g, IEEE802.11n, IEEE802.11p and IEEE802.11ac [3].

These technologies use significantly different physical layers which, from the user point of view, make them differ in terms of the bandwidth that they provide. Typically, Wi-Fi enabled devices have coverage distances ranging from 50 to more than 100 meters. In practice, this coverage distance depends greatly on the nature of the antenna and on the environment in which the devices evolve.

The IEEE802.11 protocol family relies on the same MAC layer, which is based on CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Briefly, CSMA is a probabilistic network control protocol in which a node verifies the absence of traffic (it senses the carrier) before transmitting on the shared physical medium, particularly the electromagnetic spectrum. Collision detection is aimed at improving CSMA performance by interrupting a transmission as soon as a collision is observed [3].

1.4.2.1 IEEE802.11a

IEEE802.11a uses Orthogonal Frequency Division Multiplexing (OFDM). It is the only wireless radio technology that works in the 5GHz band. The main idea behind OFDM is that since low-rate modulations (i.e. modulations with relatively long symbols compared to the channel time characteristics) are less sensitive to multipath, it should be better to send several low rate streams in parallel than sending one high rate waveform. OFDM then works by dividing one high-speed signal carrier into several lower-speed subcarriers, which are transmitted in parallel. High-speed carriers, which are 20MHz wide, are divided into 52 sub channels, each approximately 300KHz wide. OFDM uses 48 of these sub channels for transporting data, while the four others are used for error correction [3].

OFDM delivers higher data rates and a high degree of multipath reflection reconstruction, due to its encoding scheme and error correction. The IEEE802.11a standard permits the data rate to be set at 24 Mbit/s [3].

1.4.2.2 IEEE802.11b

IEEE 802.11b uses Direct Sequence Spread Spectrum (DSSS) as the physical layer technique for the standard. DSSS uses a complex technique which consists in multiplying the data being transmitted by a noise signal. This noise signal is a pseudo-random sequence of 1 and −1 values, at a frequency much higher than the original signal. The resulting signal wave looks much like white noise. This white noise can be filtered at the receiving end to recover the

original data. This filtering happens by again multiplying the same pseudo-random sequence by the received signal (because $1 \times 1 = 1$, and $-1 \times -1 = 1$). This process, known as "de-spreading", mathematically constitutes a correlation of the transmitted pseudo-random sequence with the receiver's assumed sequence. For allowing de-spreading to work correctly, the transmitted and received sequences must be synchronized. So far, IEEE 802.11b is the implementation of the IEEE 802.11 standard that has been most heavily studied in the framework of mobile Ad-hoc networks [3].

1.4.2.3 IEEE802.11g

IEEE802.11g, just like IEEE802.11a, uses orthogonal frequency-division multiplexing (OFDM), it then utilizes similar bandwidths. But unlike IEEE802.11a, IEEE802.11g works in the 2.4 GHz band. Since the draft 802.11g standard combines fundamental features from both 802.11a and 802.11b, it leads to the development of devices that can inter-operate with technologies based on both previous versions of the specification [3].

1.4.2.4 IEEE802.11n

802.11n (also sometimes known as "Wireless N") was designed to improve on 802.11g in the amount of bandwidth supported by utilizing multiple wireless signals and antennas (called MIMO technology) instead of one.

Industry standards groups ratified 802.11n in 2009 with specifications providing for up to 300 Mbps of network bandwidth. 802.11n also offers somewhat better range over earlier Wi-Fi standards due to its increased signal intensity, and it is backward-compatible with 802.11b/g gear. Fastest maximum speed and best signal range; more resistant to signal interference from outside sources [47].

1.4.2.5 IEEE802.11ac

The newest generation of Wi-Fi signaling in popular use, 802.11ac utilizes dual-band wireless technology, supporting simultaneous connections on both the 2.4 GHz and 5 GHz Wi-Fi bands. 802.11ac offers backward compatibility to 802.11b/g/n and bandwidth rated up to 1300 Mbps on the 5 GHz band plus up to 450 Mbps on 2.4 GHz [47].

1.4.3 Bluetooth

Bluetooth was initially restricted to an operating distance of just 10 meters and a speed of approximately 1 Mbit/s. When Bluetooth devices come within range of each other, they establish contact and form a temporary network called a Personal Area Network (PAN). In the Bluetooth terminology, this is also known as a Pico net. A multi-hop Ad-hoc network formed by the interaction of Bluetooth devices is called a scatter net. When using Bluetooth, the devices must establish a network session before being able to transmit any data.

Bluetooth uses the Frequency-Hopping Spread Spectrum (FHSS) technique. Unlike IEEE802.11 which establishes a communication link on a certain frequency (a channel), FHSS breaks the data down into small packets and transfers it on a wide range of frequencies across the available frequency band [3].

1.4.4 HiperLAN

The HiperLAN standard is very close to 802.11a/g in terms of the physical layers. It uses OFDM technology which differs at the MAC level and in the way the data packets are formed and devices is addressed. HiperLAN operates by sharing the 20MHz channels in the 5GHz spectrum in time, using Time Division Multiple Access (TDMA) to provide QoS. It supports two basic modes of operation: centralized mode and direct mode. The centralized mode is used in the cellular networking topology where each radio cell is controlled by an access point covering a certain geographical area. The direct mode is used in the Ad-hoc networking topology - mainly in the context of Personal Area Networking – where a radio cell covers the whole serving area. In this mode, mobile terminals in a single-cell home “network” can directly exchange data [3].

1.4.5 ZigBee

ZigBee-enabled devices conform to the IEEE802.15.4-2003 standard. This standard specifies its lower protocol layers, the physical layer (PHY), and the medium access control (MAC). It targets Low-Rate Wireless Personal Area Network (WPAN). ZigBee was intended to operate in contexts in which both Wi-Fi and Bluetooth are not suitable.

ZigBee operates in the unlicensed 2.4 GHz, 915 MHz and 868 MHz ISM bands. It uses direct-sequence spread spectrum (DSSS) coding. This makes the data rate to reach 250 Kbps per

channel in the 2.4 GHz band, 40 Kbps per channel in the 915 MHz band, and 20 Kbps in the 868 MHz band. The maximum output power of ZigBee antennas being generally 1mW, the transmission range of ZigBee nodes is between 10 and 75 meters. Observations have shown that the transmission range is heavily dependent on the environment [3].

1.5 MOBILE AD-HOC NETWORK APPLICATIONS

With the increase of portable devices as well as progress in wireless communication, Ad-hoc networking is gaining importance with the increasing number of widespread applications. Ad-hoc networking can be applied anywhere where there is little or no communication infrastructure or the existing infrastructure is expensive or inconvenient to use. Ad-hoc networking allows the devices to maintain connections to the network as well as easily adding and removing devices to and from the network. The set of applications for MANETs is diverse, ranging from large-scale, mobile, highly dynamic networks, to small, static networks that are constrained by power sources. Besides the legacy applications that move from traditional infrastructure environment into the Ad-hoc context, a great deal of new services can and will be generated for the new environment [1]. Typical applications include:

1) Military battlefield: Military equipment now routinely contains some sort of computer equipment. Ad-hoc networking would allow the military to take advantage of common place network technology to maintain an information network between the soldiers, vehicles, and military information headquarters. The basic techniques of Ad-hoc network came from this field [1].

2) Commercial sector: Ad-hoc can be used in emergency/rescue operations for disaster relief efforts, e.g. in fire, flood, or earthquake. Emergency rescue operations must take place where non-existing or damaged communications infrastructure and rapid deployment of a communication network is needed. Information is relayed from one rescue team member to another over a small handheld. Other commercial scenarios include e.g. ship-to-ship Ad-hoc mobile communication, law enforcement, etc. [1].

3) Local level: Ad-hoc networks can autonomously link an instant and temporary multimedia network using notebook computers or palmtop computers to spread and share information among participants at e.g. conference or classroom. Another appropriate local level application might be in home networks where devices can communicate directly to exchange information.

Similarly, in other civilian environments like taxicab, sports stadium, boat and small aircraft, mobile Ad-hoc communications will have many applications [1].

4) Personal Area Network (PAN): Short-range MANET can simplify the inter communication between various mobile devices (such as a PDA, a laptop, and a cellular phone) [1].

1.6 TOPOLOGY MANAGEMENT IN MOBILE AD-HOC NETWORKS

The topology of an Ad-hoc network plays a key role in the performance of the control algorithms used in the network for such purposes as scheduling of transmissions, routing, and broadcasting. In many cases, several network links are not needed for establishing efficient sharing of the channel among neighboring nodes or the routing of data packets, [5]. There are two approaches to topology management in Ad-hoc networks: power control and hierarchical topology organization [6]. Power control mechanisms adjust the power on a per-node basis, so that one-hop neighbor connectivity is balanced and overall network connectivity is ensured.

Wireless Ad-hoc network topology includes ring, mesh, star, fully connected, line, tree and bus. The most practical topology in use is the mesh topology due to its multi paths in the network [7].

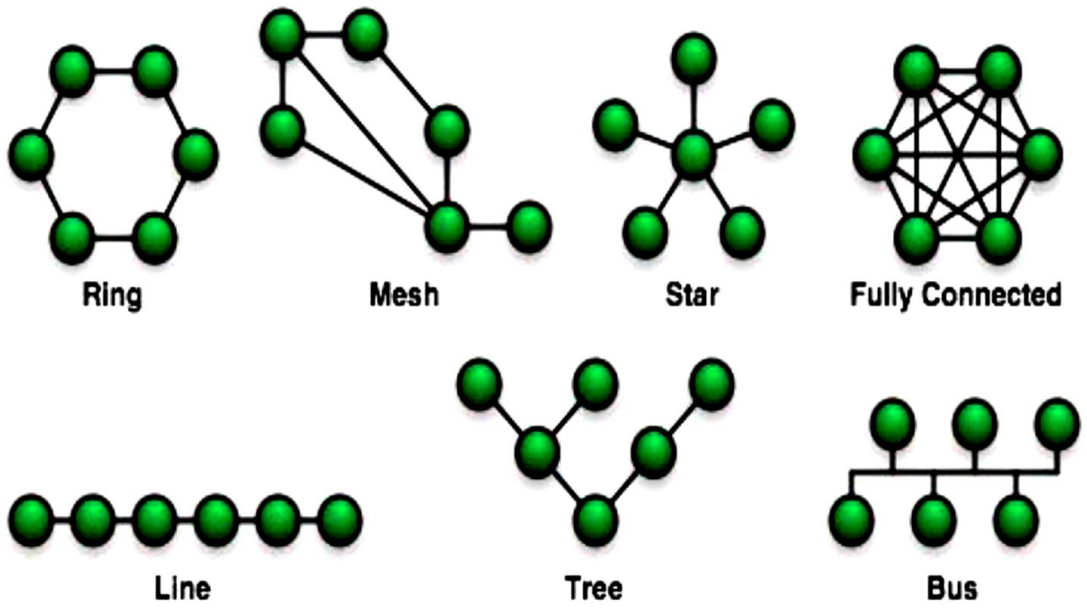


Figure 4: Mobile Ad-hoc Network Topology Examples [7]

The real attraction of a mesh topology is its ability to significantly improve the capability of any RF modulation scheme by [8]:

- Requiring less power
- Having the ability to be rapidly deployed and reconfigured
- Providing better frequency reuse
- Not requiring Lines of Sight (LOS)
- Being able to load balance around congestion
- Providing redundancy to deal with failure
- Requiring less centralized infrastructure
- Providing capacity and scalability improvements

1.7 THE OSI LAYER IN MOBILE AD-HOC NETWORK

Network protocols links to the OSI (Open Systems Interconnection) reference model, and represents the totality of protocol definitions and provide international standardization of many aspects of computer-to-computer communication.

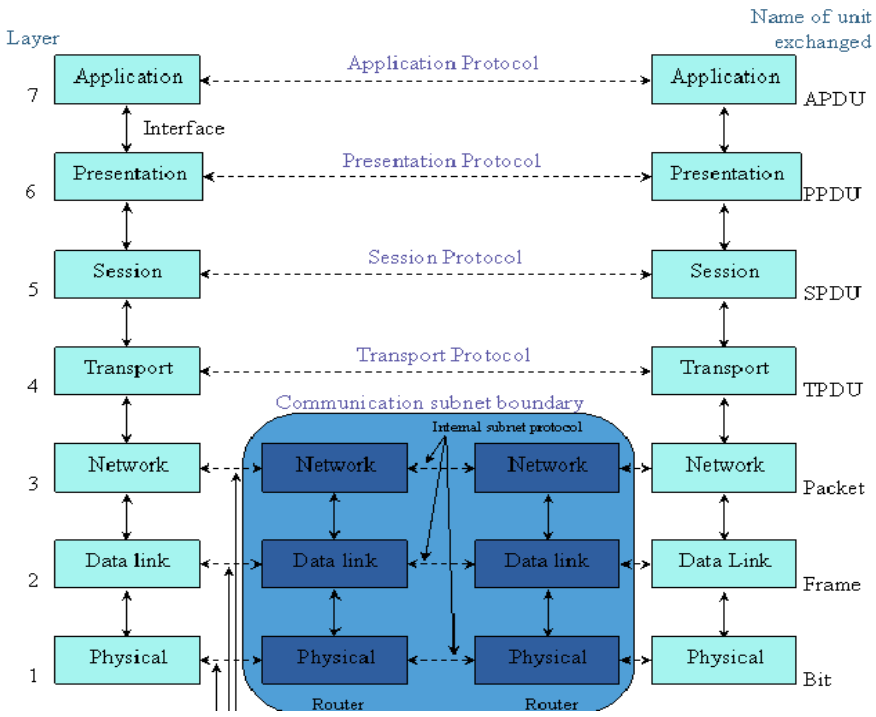


Figure 5: OSI Layer in Mobile Ad-hoc Network [9]

The seven OSI layers in mobile Ad-hoc network include [9]:

Layer 1-Physical layer: Physical medium control, bit transmission and timing.

Layer 2-Data link layer: Divided into MAC and Logical sub layer: Link control and data transmission.

Layer 3-Network layer: Network addressing and routing.

Layer 4-Transport layer: Last chance to correct network errors.

Layer 5-Session layer: Applications synchronization and connection management.

Layer 6-Presentation layer: Conversion between different machine representations.

Layer 7-Application layer: High level application support tools.

1.7.1 NETWORK LAYER REQUIREMENTS IN MOBILE AD-HOC NETWORK

The network layer in Mobile Ad-hoc Networks has several requirements. Some of these requirements are analogous to mechanisms provided in traditional wired networks. For instance, translation from hostnames to network level addresses (e.g. DNS), resolving network level addresses for link level addresses (e.g. ARP), dynamic address allocation (e.g. DHCP), etc. However, many network level requirements are specific to MANETs and result from the unique environment created by their mobility and Ad-hoc nature. The network layer in Mobile Ad-hoc Networks must provide the following capabilities:

a. Network Initiate: allow a node or nodes to establish a network.

b. Node Join: allow nodes to join the network.

c. Node Separate: allow nodes to separate from or leave the network.

d. Network Combine: allow two previously independent MANETs to combine and form one MANET.

e. Network Attach: allow a MANET to attach to an existing wired network [10].

Each of these requirements stems from the Ad-hoc and mobile characteristics of MANETs that are not present in wired networks. Fixed or wired networks require similar services, such as the Node Join listed above, but the details required by MANETs do not allow similar solutions as those traditionally provided. Dynamic host address assignment for example occurs in wired networks via the Dynamic Host Configuration Protocol (DHCP) [10].

This service requires a server to provide address assignment, leasing, aging, and reclamation. MANET nodes may have a much more Ad-hoc and transitory nature that does not allow for set server assignment to provide services. A primary goal then is to provide services to MANETs that are realized by distributed and equally shared responsibilities for each node. This creates an environment where all nodes are truly peers [10].

1.7.2 VARIABLE LENGTH ADDRESSING

Mobile Ad-hoc Networks consist of mobile nodes which will very often have significant resource constraints. Applying current network layer addressing such as traditional 32 bit IP addresses or the expanded 128 bit IPv6 addresses is possible, but results in needless overhead. Significant savings can be realized by using only the minimum number of bits required to uniquely address all the nodes participating in the current MANET. The address length can be expanded to accommodate new nodes as required. Changing the length of network layer addresses "on the fly" will require all nodes currently participating in the MANET to take some sort of action. Thus, there is a non-trivial cost associated with this. If every node that joined a MANET required reconfiguring the address length, there would be unreasonable amount of overhead required just to maintain the addresses. The two important aspects of variable length addressing are therefore the starting address length, and the increment each time the address length increases. On one end of the spectrum, the starting address length could be one bit, and the increment could be one bit. This combination would require frequent address length reconfigurations that affect all nodes currently in the network. On the other end of this spectrum, the starting address length and increment could be quite large requiring very infrequent address length changes. If these parameters are too large, then very little savings are realized by using minimal length addresses. A good compromise is to use a starting address of four bits (nibble), and increment the address space by four bits every time an address space expansion is required, [10].

a. Network Initiate

Any node can initiate a Mobile Ad-hoc Network. Initiation is accomplished in much the same way as joining a network. When a standalone node wishes to participate in a network, a join beacon is transmitted. If no reply is received in a short amount of time, then the node retransmits the join beacon and waits a longer period. This continues using an exponential back-off for several retries. If no reply is received after an appropriate number of retries, then the node assumes that they are the only node in the area wishing to participate, and they become a one node network. This node is now responsible for replying to any join beacons it detects [10].

b. Node Join

When a node wishes to participate in a network, it transmits a join beacon and waits for a response. If no reply is received in a short amount of time, the node retransmits the join beacon

and waits a longer period. This continues as above if a reply is never received. If an existing MANET is within transmission range of this node, then the following things must take place. While participating in a MANET, all nodes are responsible for supporting the network infrastructure. This includes things such as routing, and replying to any join beacons that are detected. If a join beacon is detected, then one of the receiving nodes must become the joining node's attachment agent. This requires that all the nodes which receive the join request reply to the joining agent and offer to be the attachment agent. The joining node must acknowledge only one node to act as the attachment agent [10].

c. Node Separate

Normally, node separation from an Ad-hoc network can be a passive operation that does not require any action on the part of the separating node. When this is the case, the joining of new nodes will always increase the number of addresses in use and decrease those that are available from the address space. However, this type of operation, where nodes separate from the network with no notification, does not allow the network to achieve the best possible efficiencies by using minimal length addresses. One simple solution is to require nodes that are leaving the network and have knowledge that they are leaving to send a separation message to their immediate neighbors. Any node receiving this separation message can cache the separating node's address as available and allocate it to another node that it becomes the attachment agent for. If a node tries to reuse the address on behalf of a joining node, it is still responsible to flood an address request to ensure that one of the other neighbors did not reuse the address. But in this case, the number of nodes that can reply with a negative address request is somewhat smaller than if the attachment agent had just used the next highest address. Any node receiving this address request that has the address in question in their available cache must also remove it from the cache [10].

d. Network Combine

One of the most demanding challenges to mobile Ad-hoc networks is the resolution of addresses when two networks come into contact. One solution is to ensure that every possible node be assigned a permanent network (logical) address that never changes. Simplicity is one key advantage of this approach, but it forces the use of very large addresses such as the 128 bit addresses of IPv6. The use of these large addresses results in a lot of unnecessary overhead, especially when the MANET in question is independent of a wired network and has only a few hundred, or even thousands of participants [10].

When two Mobile Ad-hoc Networks come into contact, the addresses in each can be made unique by prepending all the node addresses in one network with a high bit, and all the node addresses in the other network with a low bit. This assumes the current address lengths are equivalent, but a similar process is quite straight forward when they are not. A more formal explanation and example are included below.

If a node detects contact with another MANET by receiving a packet from another network, it must inform its own network to change addresses, and inform the source node of the foreign packet to do the same for its network. The node whose network has the highest address will automatically prepend a high bit to all their addresses, and modify their highest known address appropriately [10].

All node addresses and cached addresses (routing tables, routing caches, etc.) must be modified. The network whose node has the lower address must prepend enough low bits to all node addresses and cached addresses to make them equal in length to the new addresses from the other network, and modify the network's highest known address to be the same as the neighboring network. In the rare case that two networks happen to have the same highest known address, the network which contains the first node to recognize the incidence will claim the higher addresses and prepend a high bit, [10].

e. Network Attach

When a Mobile Ad Hoc Network comes in contact with a wired node that is able to provide gateway services to access fixed networks or the Internet, a mechanism for the MANET to attach to the fixed network should be present. This can be accomplished using the addressing scheme presented here by making three assumptions.

1) the fixed, wired network is using 128 bit IPv6 addresses,
2) there is a range of 128 bit IPv6 addresses reserved for attachment at the gateway node that are sub netted using a 64-bit network address and a 64-bit node address, and
3) the wired gateway node is able to provide minimal address translation for the MANET [10].

1.8 SELECTING BETWEEN INFRASTRUCTURE AND AD-HOC MODES

Devices in a wireless network are set up to either communicate indirectly through a central place an access point or directly, one to the other. The first is called "Infrastructure Mode" and the other is called "Ad-hoc" mode (it's also called peer-to-peer). Here are key differences between the modes.

- Because Ad-hoc Mode does not require an access point, it's easier to set up, especially in a small or temporary network.
- Infrastructure takes advantage of the high power of an access point to cover wide areas. Ad-hoc Mode connections are limited.
- Because the network topology in Ad-hoc Mode changes regularly, system resources are taken just to maintain connectivity.
- As the Ad-hoc topology changes, throughput and range will change, sometimes in unanticipated ways.
- In an Ad-hoc network with many computers, the amount of interference for all computers will go up, since each is trying to use the same frequency channel.
- In Ad-hoc Mode, chains of computers will connect to pass your data, if your computer is not directly in range. On the other hand, you do not have control over the path your data takes.

1.9 MOTIVATION FOR COOPERATIVE RETRANSMISSION

Today cooperative communication is an active area of research, and wireless devices are evolving into multipurpose systems with data extensive applications running on them, by this such applications require strong error protection and high speed connectivity. Those requirements, and along with the exploding growth of wireless networks and limited spectrum resources, have created capacity crunch and high interference in today's wireless networks. This causes in situation that a move towards the development of new wireless techniques which can achieve a more efficient use of the available spectrum or energy consumption when emerging techniques such as Multi-Input Multi-Output systems increase the spectrum efficiency in terms of the number of bits per hertz of bandwidth, because of the size, cost and power constraints posed by portable wireless devices its usage is limited. An alternative approach called cooperative communications promises to deliver some of the benefits of multi-input multi-output within the given constraints.

Cooperative communication refers to the collaborative processing and retransmission of the overheard information at those stations surrounding the source. Because of the cooperation takes full advantage of the nature of the wireless channel and create diversity, particularly transmission diversity, thereby achieving tremendous improvements in system robustness, capacity, delay, interference, and coverage range. However, the notion of cooperation is available in various forms at different protocol layers and the fundamentals of cooperative

communications lie in the physical layers to facilitate access to the physical layer information and adaptation to mobility, it is natural to introduce the notion of cooperation in the layer directly above the physical layer, namely the medium access control layer, the benefit of cooperative communication can be exploited to the maximum and evaluating the performance of these protocols have also become extremely important [11, 13].

In this study, network simulator (NS-2) is used to evaluate the performance of network protocol against different performance evaluation parameters.

1.10 COOPERATIVE TRANSMISSION

In wireless networks, cooperative transmission is used as a means to combat channel fading. In cooperative communications, relays are assigned to help a source node to deliver its information to its destination node. In addition, in wireless networks the transmitted signals that can be overheard by some nodes which do nothing but wait for their own transmission during other nodes transmission in the network.

Recently, cooperative communication schemes have been introduced to exploit the broadcasting nature of wireless transmission and many research on cooperative communication techniques have been conducted to allow stations to cooperate in their transmissions to improve the communication between all network and since transmission in the wireless channel is overheard by neighboring station which can process these signals and retransmit them to facilitate better reception. In these schemes, some overhearing nodes are also involved in the signal transmission by relaying the received signal from the source to the destination. Cooperative communications can achieve spatial diversity because signals bearing the same information go through uncorrelated channels introduced by cooperating nodes [12, 13].

Cooperative transmission can obtain spatial diversity without using multiple antennas, thus achieving more reliable transmission or consuming less power. When cooperative diversity is utilized from the Medium Access Control (MAC) layer in distributed wireless networks, three key issues need to be addressed, namely when to cooperate, whom to cooperate with and how to protect cooperative transmissions.

Since wireless channels vary with time, a source node may not always need help from a relay node, therefore it is more sensible that cooperation is only initiated when it is necessary and beneficial and secondly, one or more appropriate relay nodes need to be selected among multiple potential helpers in the network [13].

CHAPTER TWO

2. ROUTING IN MOBILE AD-HOC NETWORK

Routing is the process of finding a communication path from a source station to a destination using suitable methods and algorithms to achieve high performance in the whole network.

Since the advent of DARPA in 1970s numerous protocols have been developed for Ad-hoc mobile network. Such protocols must deal with the typical limitations of Ad-hoc network like high power consumption, low bandwidth and high error rates [3, 14].

Routing protocols can generally be classified as table driven and source- initiated on-demand driven routing protocols.

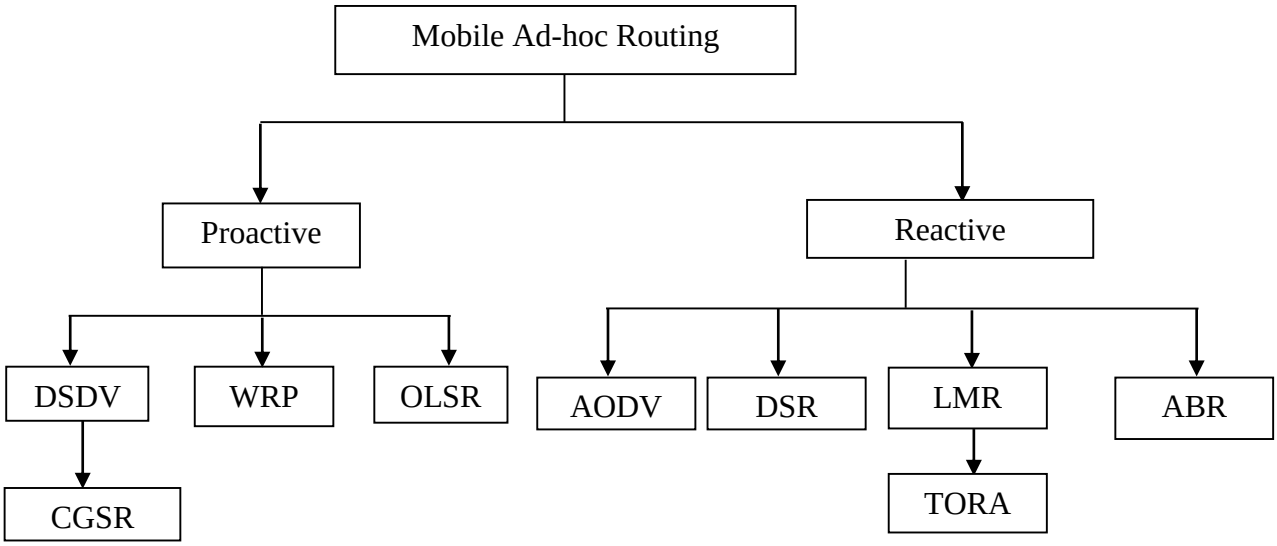


Figure 6: Routing Protocols in Mobile Ad-hoc Network [15]

2.1 PROACTIVE ROUTING PROTOCOLS

Proactive routing protocols attempt to maintain consistent and up-to-date routing information from each node to every other node in the network. These protocols require each node to maintain one or more tables to store routing information, and they respond to changes in network topology by propagating updates throughout the network to maintain a consistent network view. Some of the proactive routing protocols are listed below.

2.1.1 DESTINATION-SEQUENCED DISTANCE –VECTOR ROUTING (DSDV)

In Destination-Sequenced Distance-Vector Routing Protocol, every mobile node in the network maintains a routing table in which all the possible destinations within the network and the number of hops to each destination are recorded. Each entry is marked with a sequence

number assigned by the destination node. The sequence numbers enable the mobile nodes to distinguish stale routes from new ones, thereby avoiding the formation of routing loops. Routing table updates are periodically transmitted throughout the network to maintain table consistency [15].

2.1.2 CLUSTER HEAD GATEWAY SWITCH ROUTING (CGSR)

The Cluster head Gateway Switch Routing protocol differs from the previous protocol in the type of addressing and network organization scheme employed. Instead of a flat network, CGSR is a clustered multi hop mobile wireless network with several heuristic routing schemes i.e. by having a cluster head controlling a group of Ad-hoc nodes, a framework for code separation (among clusters), channel access, routing and bandwidth allocation can be achieved.

A cluster head selection algorithm is utilized to elect a node as the cluster head using a distributed algorithm within the cluster. The disadvantage of having a cluster head scheme is that frequent cluster head changes can adversely affect routing protocol performance since nodes are busy in cluster head selection rather than packet relaying. Hence, instead of invoking cluster head reselection every time the cluster membership changes, a Least Cluster Change (LCC) clustering algorithm is introduced. Using LCC, cluster heads only change when two cluster heads come into contact, or when a node moves out of contact of all other cluster heads.

CGSR uses DSDV as the underlying routing scheme, and hence has much of the same overhead as DSDV. However, it modifies DSDV by using a hierarchical cluster head-to-gateway routing approach to route traffic from source to destination. Gateway nodes are nodes that are within communication range of two or more cluster heads. A packet sent by a node is first routed to its cluster head, and then the packet is routed from the cluster head to a gateway to another cluster head, and so on until the cluster head of the destination node is reached. The packet is then transmitted to the destination [15].

2.1.3 WIRELESS ROUTING PROTOCOL (WRP)

The Wireless Routing Protocol (WRP) is a table-based protocol with the goal of maintaining routing information among all nodes in the network. Each node in the network is responsible for maintaining four tables: (a) distance table, (b) routing table, (c) link-cost table, and (d) message retransmission list (MRL) table. Each entry of the MRL contains the sequence number of the update message, a retransmission counter, an acknowledgment-required flag vector with

one entry per neighbor, and a list of updates sent in the update message. The MRL records which updates in an update message need to be retransmitted and which neighbors should acknowledge the retransmission.

Mobiles inform each other of link changes using update messages. An update message is sent only between neighboring nodes and contains a list of updates (the destination, the distance to the destination, and the predecessor of the destination), as well as a list of responses indicating which mobiles should acknowledge (ACK) the update. Mobiles send update messages after processing updates from neighbors or detecting a change in a link to a neighbor. In the event of the loss of a link between two nodes, the nodes send update messages to their neighbors. The neighbors then update their distance table entries and check for new possible paths through other nodes. Any new paths are relayed back to the original nodes so that they can update their tables accordingly. Nodes learn of the existence of their neighbors from the receipt of acknowledgments and other messages. If a node is not sending messages, it must send a hello message within a specified time to ensure connectivity. Otherwise, the lack of messages from the node indicates the failure of that link; this may cause a false alarm. When a mobile receives a hello message from a new node, that new node is added to the mobile's routing table, and the mobile sends the new node a copy of its routing table information [15].

2.2 REACTIVE ROUTING PROTOCOLS

A different approach from proactive routing is source initiated on demand routing. This type of routing creates routes only when desired by the source node. When a node requires a route to destination, it initiates a route discovery process within the network. This process is completed once a route is found or all possible route permutations have been examined. Once a route has been established it is maintained by some form of route maintenance procedure until either the destination becomes inaccessible along every path from the source or until the route is no longer desired. Source initiated on demand routing includes the following protocols.

2.2.1 DYNAMIC SOURCE ROUTING (DSR)

The Dynamic Source Routing (DSR) protocol is an on-demand routing protocol that is based on the concept of source routing. Mobile nodes are required to maintain route caches that contain the source routes of which the mobile is aware. Entries in the route cache are continually updated as new routes are learned.

The protocol consists of two major phases: route discovery and route maintenance. When a mobile node has a packet to send to some destination, it first consults its route cache to determine whether it already has a route to the destination. If it has an unexpired route to the destination, it will use this route to send the packet. On the other hand, if the node does not have such a route, it initiates route discovery by broadcasting a route request packet. This route request contains the address of the destination, along with the source node's address and a unique identification number [15].

Each node receiving the packet checks whether it knows of a route to the destination. If it does not, it adds its own address to the route record of the packet and then forwards the packet along its outgoing links. To limit the number of route requests propagated on the outgoing links of a node, a mobile only forwards the route request if the request has not yet been seen by the mobile and if the mobile's address does not already appear in the route record.

A route reply is generated when either the route request reaches the destination itself, or when it reaches an intermediate node which contains in its route caches an unexpired route to the destination. By the time the packet reaches either the destination or such an intermediate node, it contains a route record yielding the sequence of hops taken. Figure 7 illustrates the formation of the route record as the route request propagates through the network [15].

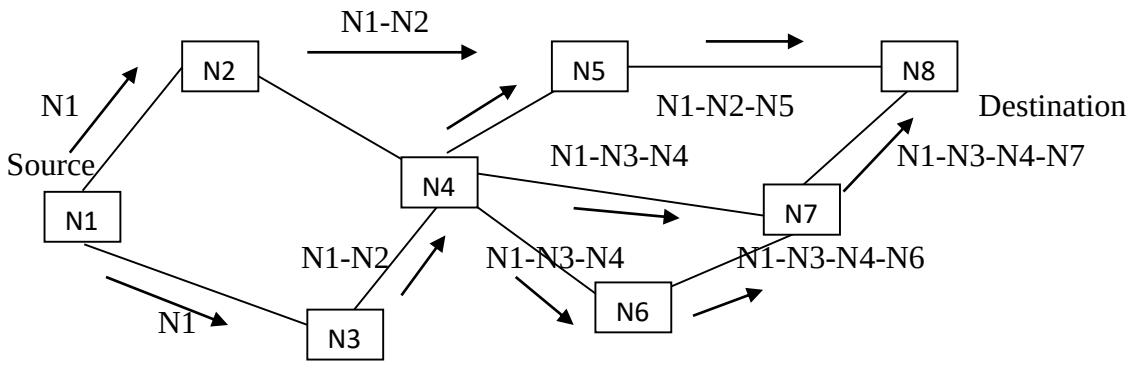


Figure 7: Route Record during Route Discovery [15]

If the node generating the route reply is the destination, it places the route record contained in the route request into the route reply. If the responding node is an intermediate node, it will append its cached route to the route record and then generate the route reply. To return the route reply, the responding node must have a route to the initiator. If it has a route to the initiator in its route cache, it may use that route. Otherwise, if symmetric links are supported, the node may reverse the route in the route record. If symmetric links are not supported, the node may

initiate its own route discovery and piggyback the route reply on the new route request. Figure 8 shows the transmission of the route reply with its associated route record back to the source node [15].

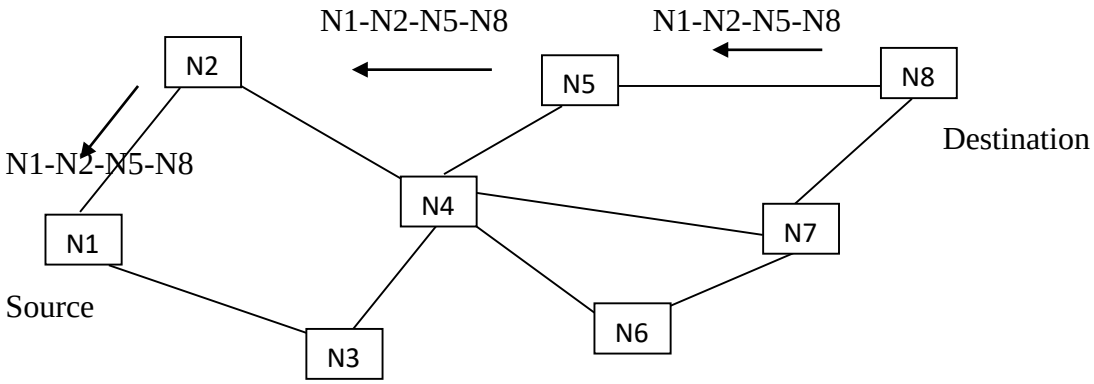


Figure 8: Propagation of Route Reply with Route Record [15]

Route maintenance is accomplished using route error packets and acknowledgments. Route error packets are generated at a node when the data link layer encounters a fatal transmission problem. When a route error packet is received, the hop in error is removed from the node's route cache and all routes containing the hop are truncated at that point. In addition to route error messages, acknowledgments are used to verify the correct operation of the route links. Such acknowledgments include passive acknowledgments, where a mobile can hear the next hop forwarding the packet along the route [15].

2.2.2 AD-HOC ON-DEMAND DISTANCE VECTOR (AODV)

The Ad-hoc On-Demand Distance Vector (AODV) algorithm enables dynamic, self-starting, multi hop routing between participating mobile nodes wishing to establish and maintain an Ad-hoc network. AODV allows mobile nodes to obtain routes quickly for new destinations, and does not require nodes to maintain routes to destinations that are not in active communication. AODV allows mobile nodes to respond to link breakages and changes in network topology in a timely manner. The operation of AODV is loop-free, and by avoiding the Bellman-Ford "counting to infinity" problem offers quick convergence when the Ad-hoc network topology changes (typically, when a node moves in the network). When links break, AODV causes the affected set of nodes to be notified so that they can invalidate the routes using the lost link.

One distinguishing feature of AODV is its use of a destination sequence number for each route entry. The destination sequence number is created by the destination to be included along with any route information it sends to requesting nodes. Using destination sequence numbers ensures loop freedom and is simple to program. Given the choice between two routes to a

destination, a requesting node is required to select the one with the greatest sequence number. Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs) are the message types defined by AODV. These message types are received via UDP, and normal IP header processing applies. So, for instance, the requesting node is expected to use its IP address as the Originator IP address for the messages. For broadcast messages, the IP limited broadcast address (255.255.255.255) is used. This means that such messages are not blindly forwarded. However, AODV operation does require certain messages (e.g., RREQ) to be disseminated widely, perhaps throughout the Ad-hoc network. The range of dissemination of such RREQs is indicated by the TTL in the IP header. Fragmentation is typically not required [16].

As long as the endpoints of a communication connection have valid routes to each other, AODV does not play any role. When a route to a new destination is needed, the node broadcasts a RREQ to find a route to the destination. A route can be determined when the RREQ reaches either the destination itself, or an intermediate node with a 'fresh enough' route to the destination. A 'fresh enough' route is a valid route entry for the destination whose associated sequence number is at least as great as that contained in the RREQ. The route is made available by unicasting a RREP back to the origination of the RREQ. Each node receiving the request caches a route back to the originator of the request, so that the RREP can be unicast from the destination along a path to that originator, or likewise from any intermediate node that can satisfy the request [16].

2.2.3 TEMPORALLY-ORDERED ROUTING ALGORITHM (TORA)

TORA (Temporally-Ordered Routing Algorithm) is a highly adaptive, loop-free, distributed routing algorithm based on the concept of link reversal. TORA is proposed to operate in a highly dynamic mobile networking environment. It is source-initiated and provides multiple routes for any desired source/destination pair. The key design concept of TORA is the localization of control messages to a very small set of nodes near the occurrence of a topological change. To accomplish this, nodes need to maintain routing information about adjacent nodes. The protocol performs three basic functions: (a) route creation, (b) route maintenance, and (c) route erasure. During the route creation and maintenance phases, nodes use a height metric to establish a directed acyclic graph (DAG) rooted at the destination. Thereafter, links are assigned a direction (upstream or downstream) based on the relative height metric of neighboring nodes [16, 17].

Timing is an important factor for TORA because the height metric is dependent on the logical time of a link failure; TORA assumes all nodes have synchronized clocks (accomplished via an external time source such as Global Positioning System). TORA's metric is a quintuple comprised of five elements, namely: (a) logical time of a link failure, (b) the unique ID of the node that define the new reference level, (c) a reflection indicator bit, (d) a propagation ordering parameter, and (e) the unique ID of the node. The first three elements collectively represent the reference level. A new reference level is defined each time a node loses its last downstream link due to a link failure. TORA's route erasure phase essentially involves flooding a broadcast "clear packet" (CLR) throughout the network to erase invalid routes [16].

In TORA, there is a potential for oscillations to occur, especially when multiple sets of coordinating nodes are concurrently detecting partitions, erasing routes, and building new routes based on each other. Because TORA uses inter nodal coordination, its instability problem is like the "count-to-infinity" problem in distance-vector routing protocols, except that such oscillations are temporary and route convergence will ultimately occur.

2.2.4 ASSOCIATIVITY-BASED ROUTING (ABR)

The Associativity-Based Routing (ABR) protocol is free from loops, deadlock, and packet duplicates, and defines a new routing metric for Ad-hoc mobile networks. This metric is known as the degree of association stability. In ABR, a route is selected based on the degree of association stability of mobile nodes. Each node periodically generates a beacon to signify its existence. When received by neighboring nodes, this beaconing causes their associativity tables to be updated. For each beacon received, the associativity tick of the current node with respect to the beaconing node is incremented. Association stability is defined by connection stability of one node with respect to another node over time and space. A high degree of association stability may indicate a low state of node mobility, while a low degree may indicate a high state of node mobility [16].

Associativity ticks are reset when the neighbors of a node or the node itself moves out of proximity. A fundamental objective of ABR is to derive longer-lived routes for Ad-hoc mobile networks.

The three phases of ABR are: (a) route discovery, (b) route re-construction (RRC), and (c) route deletion. The route discovery phase is accomplished by a broadcast query and await-reply (BQ-REPLY) cycle. A node desiring a route broadcasts a BQ message in search of mobiles

that have a route to the destination. All nodes receiving the query (that are not the destination) append their addresses and their associativity ticks with their neighbors along with QoS information to the query packet. A successor node erases its upstream node neighbors' associativity tick entries and retains only the entry concerned with itself and its upstream node. In this way, each resultant packet arriving at the destination will contain the associativity ticks of the nodes along the route to the destination. The destination is then able to select the best route by examining the associativity ticks along each of the paths. In the case where multiple paths have the same overall degree of association stability, the route with the minimum number of hops is selected. The destination then sends a REPLY packet back to the source along this path. Nodes propagating the REPLY mark their routes as valid. All other routes remain inactive and the possibility of duplicate packets arriving at the destination is avoided [17].

Route re-construction may consist of partial route discovery, invalid route erasure, valid route updates, and new route discovery, depending on which node(s) along the route move. Movement by the source results in a new BQ-REPLY process. The RN [15] message is a route notification that is used to erase the route entries associated with downstream nodes. When the destination node moves, the immediate upstream node erases its route and determines if the node is still reachable by a localized query (LQ [H]) process, where H refers to the hop count from the upstream node to the destination. If the destination receives the LQ packet, it REPLYs with the best partial route; otherwise, the initiating node times out and the process backtracks to the next upstream node. Here an RN message is sent to the next upstream node to erase the invalid routes and inform this node it should invoke the LQ [H] process. If this process results in backtracking more than halfway to the source, the LQ process is discontinued and a new BQ process is initiated at the source [16,17].

CHAPTER THREE

3. DATA LINK LAYER

Data Link Layer is one of the seven OSI layers of mobile Ad-hoc networks. The main functions of the data link layer are data link control and media access control. Data link control is deals with the design and procedures for communication between two adjacent nodes: node to node communication. The other functions of the data link layer are media access control, regarding how to share the links in the network [18].

We can consider the data link layer as two sub-layers; the upper sublayer called Logical link control (LLC), is responsible for data link control, and the lower sub-layer that is called Media Access Control (MAC) is responsible for resolving access to the shared media. If the channel is dedicated, we may not need the lower sublayer, [18, 19].

When mobile nodes are connected through wireless channel they may uses a common link, called a multipoint, so that we need a multiple access control protocols to coordinate access to the link. This is similar for multipoint networks and many formal protocols have been devised to handle access to a shared link, [18, 19].

3.1 MEDIUM ACCESS CONTROL

The IEEE 802.11 MAC layer covers three functional areas: reliable data delivery, access control, and security, [20, 21]. The MAC sub-layer is also responsible for the channel access procedures, protocol data unit (PDU) addressing, frame formatting, error checking, and fragmentation and reassembly of MAC Service Data Unit (MSDUs). The MSDU is the Service Data Unit that is received from the Logical Link Control sub-layer which lies above MAC sub-layer in a protocol stack. The Logical Link Control (LLC) and MAC sub-layer are collectively referred to as the Data Link Layer [19, 22, 23, 24].

RELIABLE DATA DELIVERY

A wireless LAN using the IEEE 802.11 physical and MAC layers is subject to considerable unreliability. Noise, interference, and other propagation effects result in the loss of a significant number of frames and a few MAC frames may not successfully be received. This situation can be dealt with by reliability mechanisms at a higher layer, such as transport layer. There is not a good option to use timers for retransmission at higher layers. It is therefore more efficient to deal with errors at the MAC level. For this purpose, IEEE 802.11 includes a frame exchange

protocol in such away when a node receives a data frame from another node; it returns an acknowledgment (ACK) frame to the source node. This exchange is treated as an atomic unit, not to be interrupted by a transmission from any other nodes. If the source does not receive an ACK frame within a predefined short period, either its data frame or returning ACK frame was damaged, thus the source retransmits the frame [20, 25].

The basic data transfer mechanism in IEEE 802.11 involves an exchange of control frames. A source first issues a Request to Send (RTS) frame to the destination. The destination then responds with a Clear to Send (CTS). After receiving the CTS, the source transmits the data frame, and the destination responds with an ACK. The RTS alerts all stations that are within reception range of the source that an exchange is under way; these stations hold back from transmission to avoid a collision between two frames transmitted at the same time. In similarly ways, the CTS alerts all stations that are within reception range of the destination that an exchange is under way. The RTS/CTS portion of the exchange is a required function of the MAC but may be disabled [20, 23, 25].

Distributed access protocols, which, like Ethernet, distribute the decision to transmit over all the nodes using a carrier sense mechanism; and centralized access protocols, which involve regulation of transmission by a centralized decision maker. A distributed access protocol makes sense for an Ad-hoc network of peer workstations and may also be attractive in other wireless LAN configurations that consist primarily of burst traffic. A centralized access protocol is natural for configurations in which several wireless stations are interconnected with each other and some sort of base station that attaches to a backbone wired LAN; it is especially useful if some of the data is time sensitive or high priority [20, 23, 25].

Figure 9 illustrates the MAC layer in IEEE 802.11 standard. The lower sub-layer of the MAC layer is the distributed coordination function (DCF) which uses a contention algorithm to provide access to all traffic and asynchronous traffic directly uses DCF. The point coordination function (PCF) is a centralized MAC algorithm used to provide contention-free service. PCF is built on top of DCF and exploits features of DCF to assure access for its users [23, 26].

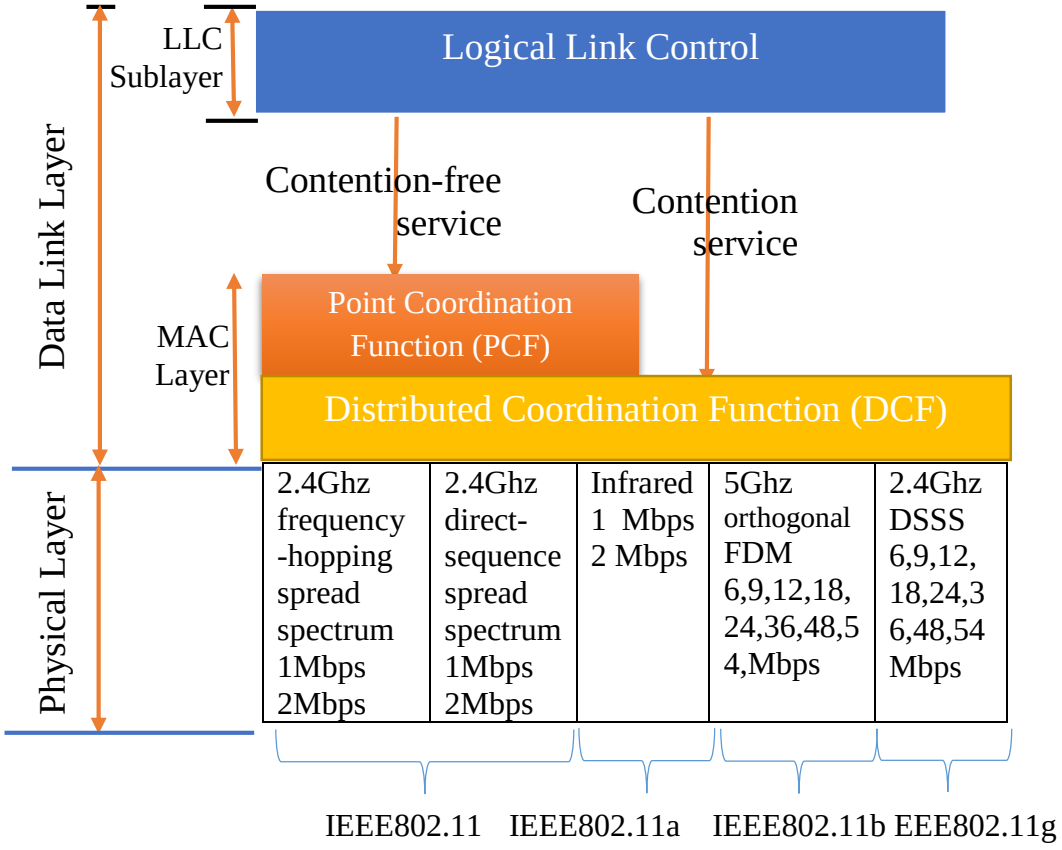


Figure 9: MAC Layer in IEEE 802.11 Standard [26]

DISTRIBUTED COORDINATION FUNCTION

The DCF sub-layer makes use of a simple carrier sense multiple access (CSMA) algorithm. The stations always listen to the medium when they have a MAC frame to transmit. If the medium is idle, the station may transmit; otherwise all stations must wait until the current transmission is complete before transmitting. The distributed coordination function does not include a collision detection function for example carrier sense multiple access with collision detection (CSMA/CD) because collision detection is not practical on a wireless network. A transmitting station cannot effectively distinguish incoming weak signals from noise and the effect of its own transmission when the dynamic range of the signals on the medium is very large. DCF includes a set of delays that amounts to a priority scheme to ensure the smooth and fair function of algorithm and start by considering a single delay known as an inter-frame space (IFS). There are three different IFS values and the algorithm is best explained by initially ignoring this detail which using IFS, the rules for CSMA access are as follows.

If a station has a frame to transmit senses the medium and if the medium is idle, it waits for a short period to see if the medium remains idle for a time equal to IFS. If so, the station may transmit immediately.

If the medium is busy and the current transmission is not over or because the medium becomes busy during the IFS idle time, the station defers transmission and continues to monitor the medium until the current transmission is over. The station must delay other IFS if the current transmission is over, once the medium remains idle for this period, then the station backs off a random amount of time and again senses the medium. If the medium is still idle, the station may transmit. If the medium becomes busy, during the back off time, the back off timer is halted and resumes when the medium becomes idle.

In situation when the transmission is unsuccessful, which is determined by the absence of an acknowledgement, then it is assumed that a collision has occurred. Binary exponential back off provides a means of handling a heavy load. Repeated failed attempts to transmit result in longer and longer back off times, which causes to smooth out the load without such a back off, the following situation could occur:

Collision can be happened when two or more stations attempt to transmit at the same time. These stations then immediately attempt to retransmit, causing a new collision. This situation lead to coordination and the preceding scheme is refined for distributed coordinate function (DCF) to create access priority-based scheme by the simple mean of using three values for IFS

Short Inter Frame Space (SIFS): The shortest IFS, used for all immediate response actions, [23, 24, 26]

Point Coordination Function Inter Frame Space (PIFS): A mid length IFS, used by the centralized controller in the PCF scheme when issuing polls [23, 24, 26].

Distributed Coordination Function Inter Frame Space (DIFS): The longest IFS, used as a minimum delay for asynchronous frames contending for access [23, 24, 26].

As shown in figure 10 IEEE 802.11 MAC timing (Basic access method) and the use of this time values. Here first consider the SIFS. Any station using SIFS to determine transmission opportunity has, in effect, the highest priority, because it will always gain access in preference to a station waiting an amount of time equal to PIFS or DIFS. The SIFS is used in the following circumstances: [23, 24, 26].

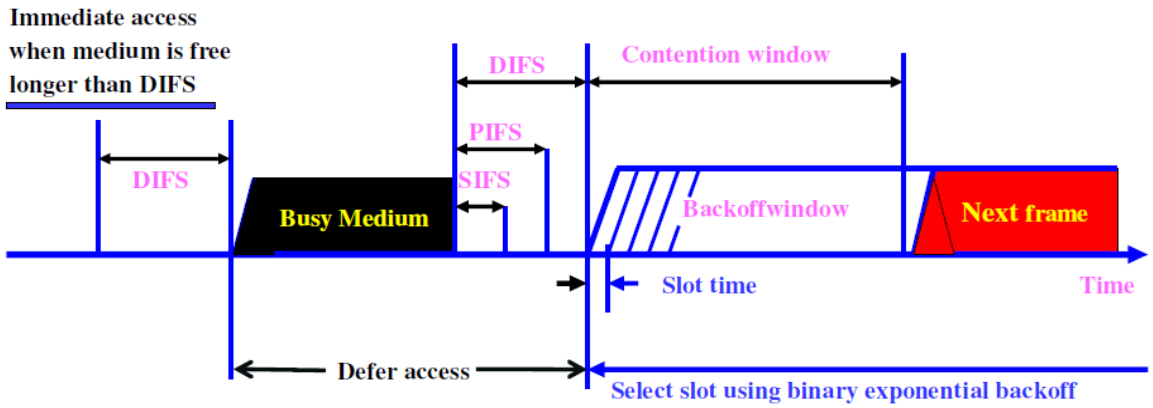


Figure 10: IEEE 802.11 MAC timing (Basic access method) [26, 27]

Acknowledgment (ACK): When a station receives, a frame addressed only to itself not multicast or broadcast, it responds with an ACK frame after waiting only for an SIFS gap [23, 24, 26, 27].

Clear to Send (CTS): A station can ensure that its data frame will get through by first issuing a small Request to Send (RTS) frame. The station to which this frame is addressed should immediately respond with a CTS frame if it is ready to receive. All other stations receive the RTS and defer using the medium [23, 24, 26].

Poll response: The point coordinator makes use of PIFS when issuing polls.

POINT COORDINATION FUNCTION

The point coordination function (PCF) is an optional capability that can be used to provide connection-oriented, contention-free services by enabling polled stations to transmit without contending for the channel. This is an alternative access method implemented on top of the DCF is called PCF. The operation consists of polling by the centralized polling master. The point coordinator makes use of PIFS when issuing polls [27].

Because PIFS is smaller than DIFS, the point coordinator can seize the medium and lock out all asynchronous traffic while it issues polls and receives responses [23, 24, 26].

At the beginning of a super frame, the point coordinator may optionally seize control and issue polls for a given period. This interval varies because of the variable frame size issued by responding stations. The remainder of the super frame is available for contention-based access. At the end of the super frame interval, the point coordinator contends for access to the medium using PIFS. If the medium is idle, the point coordinator gains immediate access and a full super frame period follows.

The PCF function is performed by the point coordinator (PC) in the AP within a BSS. Stations within the BSS that can operate in the contention-free poll (CFP) are known as CF-aware stations.

3.2 IEEE 802.11 DCF SCHEME DESCRIPTION

The IEEE 802.11 standard for wireless networks incorporates two medium access methods, which one of two is mandatory called, Distributed Coordination Functions (DCF) method and the other is optional, Point Coordination Functions (PCF) method.

Wireless stations communicate with each other using a contention based channel access method CSMA/CA in the DCF. On the top of DCF is built PCF, which is only possible in infrastructure mode. The access-point of cell acts as a coordinator called the point coordinator (PC) for that cell. The PC grants contention free channel access to individual nodes by polling them for transmission.

The DCF is an asynchronous data transmission function and best suits delay insensitive traffic and PCF best suits delay sensitive traffic. Basically, the DCF provides an equal opportunity to access the shared wireless channel to the contending stations, this equal access opportunity results in an equal share of the bandwidth or throughput.

The DCF defines two techniques for packet transmissions. The default is a two-way handshaking mechanism called basic access method as shown in figure 11.

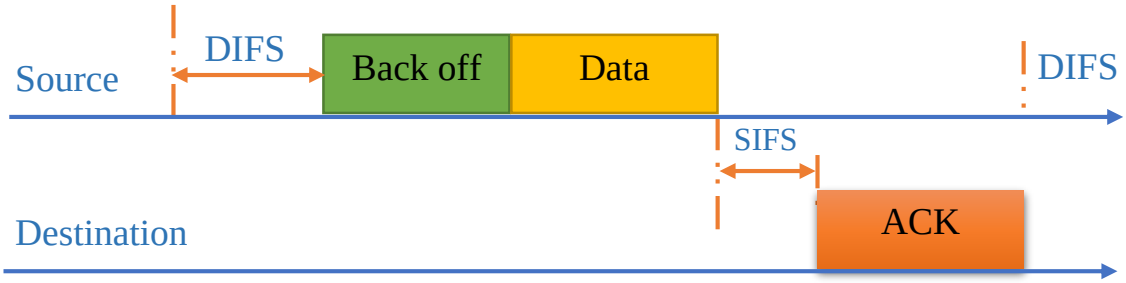


Figure 11: IEEE 802.11 DCF basic access scheme [27]

Another optional technique is a mechanism known as a Request To Send / Clear To Send (RTS/CTS) method as shown in figure 12. The use of the RTS and CTS frames can improve performance in two important ways. The hidden station problem is mitigated since a long DATA frame is transmitted only after the channel has been reserved.

Because the RTS and CTS frames are short, a collision involving an RTS or CTS frame will last only for the duration of the short RTS or CTS frame. Once the RTS and CTS frames are

correctly transmitted, the following DATA and ACK frame should be transmitted without collision.

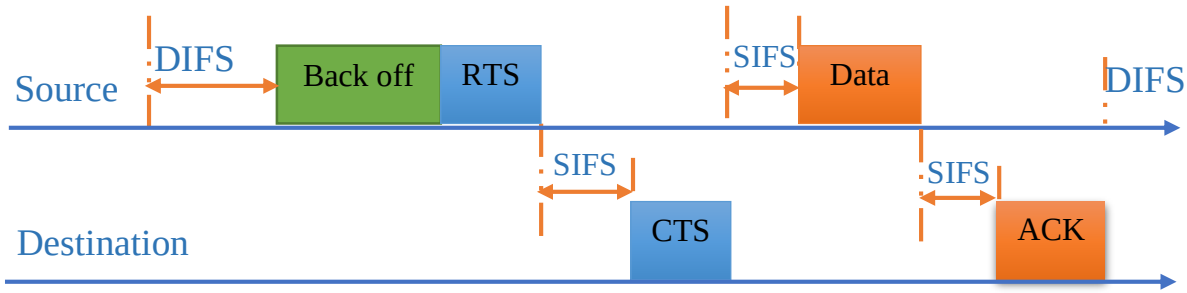


Figure 12: IEEE 802.11 DCF RTS/CTS scheme [27]

Although the RTS/CTS exchange can help reduce collisions, it also introduces delay and consumes channel resources. For this reason, the RTS/CTS exchange is only used if at all to reserve the channel for the transmission of a long DATA frame. In practice, each wireless station can set an RTS threshold such that the RTS/CTS sequence is used only when the frame is longer than the threshold. For many wireless stations, the default RTS threshold value is larger than the maximum frame length, so the RTS/CTS sequence is skipped for all DATA frames sent.

3.3 COOPERATIVE COMMUNICATION IN MAC LAYER

IEEE 802.11 WLANs provide flexible access to the Internet; have been widely researched and deployed for several years. Medium access control (MAC) protocol is one of the key components in WLAN, which employs distributed coordination function (DCF) as the primary mechanism to access the medium. Distributed coordinate function is based on the carrier sense multiple access with collision avoidance (CSMA/CA) protocol that use a binary exponential back off (BEB) algorithm to provide access to heterogeneous traffic.

3.3.1 REVIEW OF DISTRIBUTED COORDINATION FUNCTION

We briefly review the process of the basic access mode of DCF. If a station has a packet to send, it monitors the channel activity. If the channel is idle for a period equals to a distributed coordination function inter-frame space (DIFS), the station transmits. Otherwise, the station defers and monitors the channel until it is measured idle for a DIFS. At this point, the station generates a random back off interval before transmitting; this method causes to minimize the probability of collision with packets being transmitted by other stations [27].

Distributed coordinate function uses an exponential back off algorithm and the back off time that is uniformly chosen which in the range (0, $CW-1$), here the CW is called contention

window which *is* set equal to *CWmin* known as minimum contention window at the first transmission. After each unsuccessful transmission, *CW* will be doubled until it reaches the maximum value $CWmax = (2^m)W$, where W equals to *CWmin* and *m* is retrying limit (m=7 for basic access mechanism and m=4 for the RTS/CTS exchange mechanism). The back off counter is decremented as long as channel is sensed idle, and frozen when a transmission is detected on the channel, and the station transmits when the back off counter reaches zero.

In RTS/CTS based access mode, nodes transmit data using special short RTS and CTS frames prior to the transmission of the data packet in favor of shortening the collision time interval. After CTS is correctly received, the source node can transmit packet. Still there is a possibility that a collision may occur among two or more short RTS/CTS frames, however, compared to the case when large data packet may collide, less bandwidth will be wasted.

3.3.2 COOPERATION IN RTS/CTS SCHEME

Since IEEE802.11 basic scheme is used in case of small data transmission introducing cooperative communication here may not be efficient or at least increase over head. Hence, I have implemented my cooperative communication model in RTS/CTS scheme in the case when attempt for direct transmission is failed.

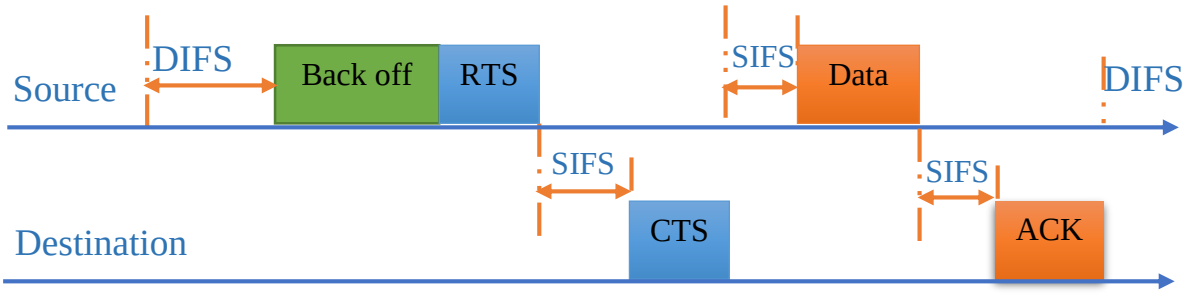
In a Wireless network topology, when not all nodes are within each other's range, carrier sense only provides information about collisions at the sender, not at the receiver. This leads to problems like "hidden terminal" (where two sender nodes out of range of each other transmit packets at the same time, to the same receiver, resulting in collisions at the receiver).

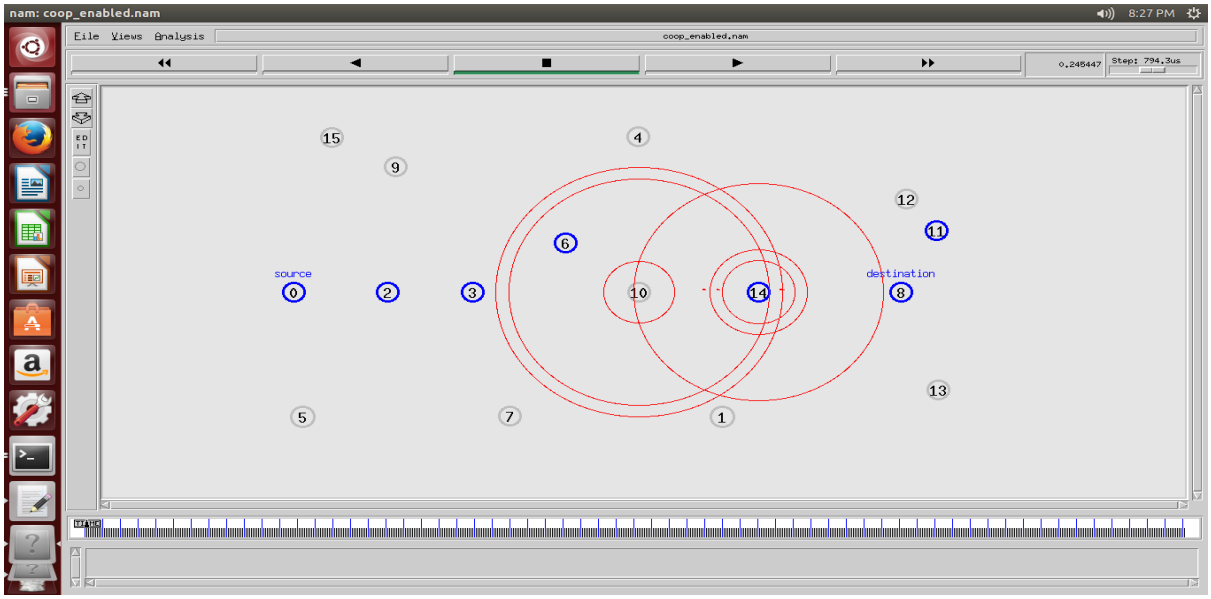
To solve such problems, control packets (RTS/CTS) are used, as an alternative to the carrier sense protocol. When a node wants to transmit packets, it first sends a RTS (Request To Send) packet to the receiver. All nodes within the sender's range receive this RTS packet. Every node hearing the RTS packet will defer transmission. If receiver is not receiving data from any other node, it responds with a CTS (Clear To Send) packet. This packet is again received by every node within receiver's range. So, all nodes in the range defer transmission. Sender node sends the data packet after receiving CTS. If sender node does not hear the CTS, it will time-out and schedule retransmission of RTS. On receiving the data packet, receiver sends an ACK to the sender. Link layer ACKs are used to facilitate recovery from collision faster. RTS and CTS packets are very short, hence their collision is very less likely [23, 27].

If RTS packets from 2 sender nodes collide at the receiver, sender nodes will not receive a CTS packet, thus the nodes will back off for certain duration, resulting in collision detection. This solves the hidden terminal problem, even though two sender nodes may be out of range of each other, they both are in the range of the receiver they want to send the data packets to. Hence, when receiver responds to one node with a CTS packet, it will be heard by the other sender node too and it will defer transmission [23].

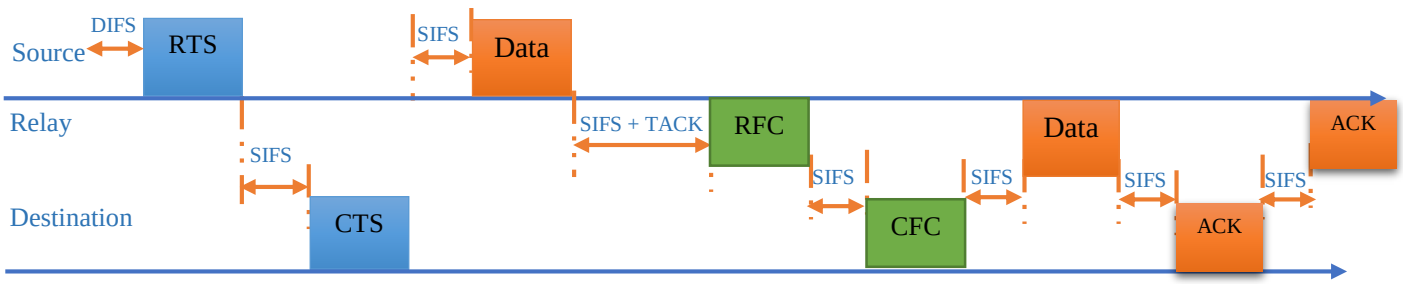
In cooperative RTS/CTS schemes packet transmission, there are three possibilities that could occurs:

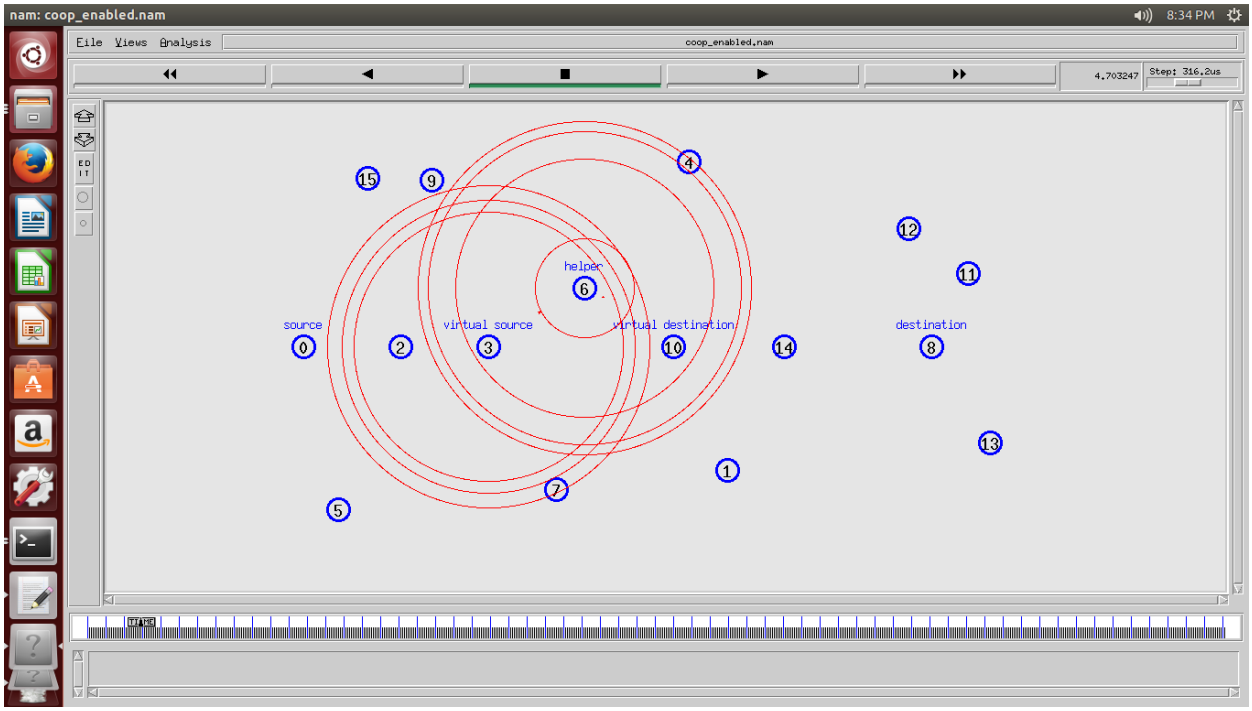
1. direct transmission attempt between Source and destination is successfully
2. cooperative transmission attempt from Relay node to Destination is successfully
3. transmission failure



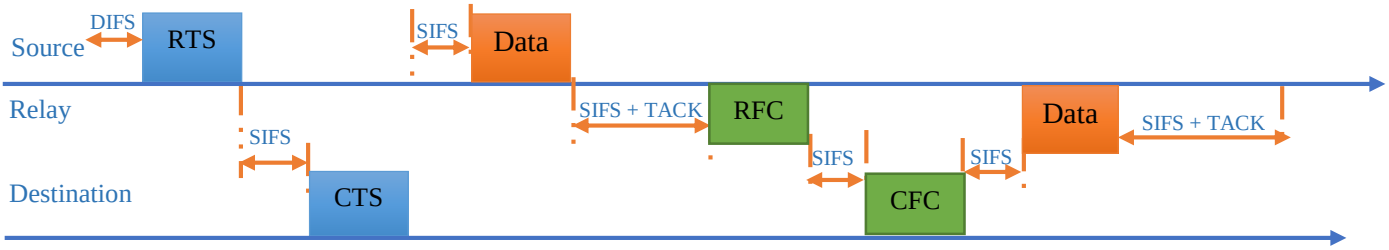


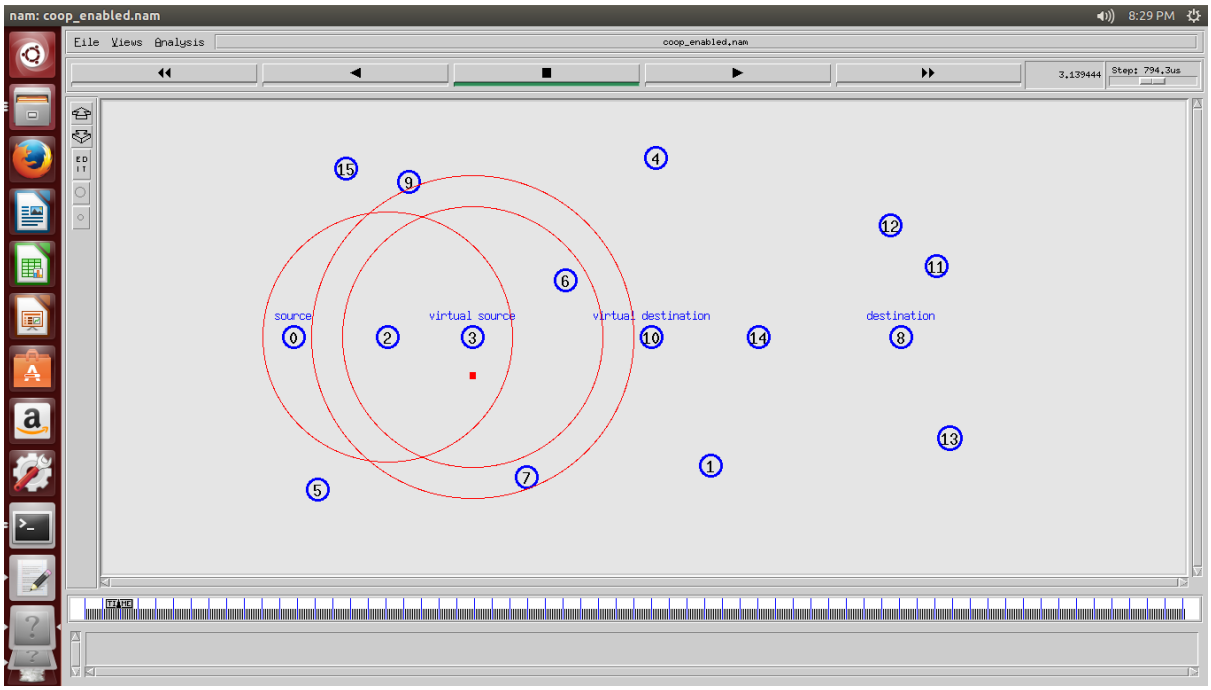
1. Direct transmission is successful





2. Cooperative transmission is successful





3. Transmission is failed

Figure 13: Cooperative communication in RTS/CTS scheme

As shown in figure 13 in RTS/CTS transmission, source and destination perform two handshakes. If the direct transmission of RTS/CTS scheme fails, after waiting for pre-specified timeout the relay node will start cooperative retransmission automatically as shown on step 2 in figure 13. To deal with the hidden terminal problem, another control frame named Request for Cooperation (RFC) is introduced in the cooperative RTS/CTS scheme. The Request For Cooperation (RFC) and Clear For Cooperation (CFC) frames contain the time duration that will be consumed by their cooperative retransmission sequences respectively to reserve the channel between relay and destination that have the same format as RTS and CTS frames and will be transmitted at the same rate as the RTS and CTS frames respectively, in order to protect the cooperative transmission packet exchange in the sensing range.

a. WHEN TO INITIATE COOPERATION

The cooperative communication initiated when direct transmission is interrupted or failed. The reason for this failure could be due to power shortage of immediate destination node or intermediate destination is out of coverage area of the sending node.

i. The cooperation enabled after waiting for *TCTS timeout* and if not received CTS frame from the destination node.
ii. The cooperation enabled after waiting for *TACK timeout*, if not received an acknowledgment frame.

b. WHOM TO COOPERATE WITH

The potential node that have overheard the control frame transmission between source and destination. This node decides to participate in the cooperation only if it hears both RTS and CTS frames sent between source and destination but not heard ACK frame.

c. HOW TO PROTECT COOPERATION

The communication protected by the control frames like RFC and CFC frames, and by network allocation vector (NAV) by reserving the channel for predetermined and expected time by the time scheduler for handshakes between the nodes and reception of data's and acknowledgments.

CHAPTER FOUR

4. DESCRIPTION AND RELEVANCE OF DSR PROTOCOL

The Dynamic Source Routing protocol (DSR) is a simple and efficient routing protocol designed specifically for use in multi-hop wireless Ad-hoc networks of mobile nodes [14]. While using DSR, the network is completely self-organizing and self-configuring, requiring no existing network infrastructure or administration. Network nodes (computers) cooperate to forward packets for each other to allow communication over multiple "hops" between nodes not directly within wireless transmission range of one another. As nodes in the network move about or join or leave the network, and as wireless transmission conditions such as sources of interference change, all routing is automatically determined and maintained by the DSR routing protocol. Since the number or sequence of intermediate hops needed to reach any destination may change at any time, the resulting network topology may be quite rich and rapidly changing.

DSR adopts an on-demand approach of source routing to allow for the efficient forwarding of packets. This adaptable and economical method of data transfer over large areas was created and developed for mobile Ad-hoc networks (MANET) at Carnegie Mellon University. Each packet sent over DSR carries a header with an ordered source list of the nodes to traverse to reach its destination [14].

The data structure that DSR uses to keep track of the routing information is a route cache, which stores the route replies it receives from route requests. Since packets already contain the routing path, intermediate nodes do not need to maintain a current map of the network infrastructure. This also allows a packet to be loop-free, and eliminates the necessity for the frequent neighbor updates or route advertisement packets that are essential for the operation of other routing protocols. The two most prominent features of DSR are route discovery and route maintenance. Route discovery is initiated through route request packets from a sender to a destination. The request packets are broadcast in a controlled way to all nodes in the network and are answered by either the destination node or an intermediate node that knows a route to the destination. This on-demand feature allows DSR to be completely self-organizing and self-configuring, making changes only to the routes requested by sending nodes. It has proven useful in eliminating the flood of update messages due to the frequent addition or departing of nodes as in mobile wireless networks.

DSR has developed into a prominent protocol in large scale Ad-hoc mobile routing (along with AODV—Ad-hoc On-demand Distance Vector) and has become a focal point for several routing research and development projects. A notable venture in the use of DSR protocol is in an undersea systems network in Australia. In the project, researchers manipulate dynamic source routing to search for a robust and efficient set of routing methods for AUVs (autonomous undersea vehicles). Another example of the capacity of DSR could be an Ad-hoc network wireless cellular system in a developing country. Such environments with no situation where the node itself does not know it is in a covert channel transfer with another node.

4.1 OVERVIEW AND IMPORTANT PROPERTIES OF THE PROTOCOL

As stated earlier, the DSR protocol is composed of two mechanisms that work together to allow the discovery and maintenance of source routes in the Ad-hoc network [5, 14, 15]:

- ***Route Discovery*** is the mechanism by which a source node **S** wishing to send a packet to a destination node **D** obtains a source route to **D**. Route Discovery is used only when **S** attempts to send a packet to **D** and does not already know a route to **D**.
- ***Route Maintenance*** is the mechanism by which node **S** is able to detect, while using a source route to **D**, if the network topology has changed such that it can no longer use its route to **D** because a link along the route no longer works. When Route Maintenance indicates a source route is broken, **S** can attempt to use any other route it happens to know to **D**, or can invoke Route Discovery again to find a new route. Route Maintenance is used only when **S** is sending packets to **D**.

Route Discovery and Route Maintenance each operate entirely on demand. Unlike other protocols, DSR requires no periodic packets of any kind at any level within the network. For example, DSR does not use any periodic routing advertisement, link status sensing, or neighbor detection packets, and does not rely on these functions from any underlying protocols in the network. This entirely on-demand behavior and lack of periodic activity allows the number of overhead packets caused by DSR to scale all the way down to zero, when all nodes are approximately stationary with respect to each other and all routes needed for current communication have already been discovered. As nodes begin to move more or as communication patterns change, the routing packet overhead of DSR automatically scales to only that needed to track the routes currently in use.

In response to a single Route Discovery (as well as through routing information from other packets overheard), a node may learn and cache multiple routes to any destination. This allows the reaction to routing changes to be much more rapid, since a node with multiple routes to a destination can try another cached route if the one it has been using should fail. This caching of multiple routes also avoids the overhead of needing to perform a new Route Discovery each time a route in use breaks.

The operation of Route Discovery and Route Maintenance in DSR are designed to allow unidirectional links and asymmetric routes to be easily supported. In wireless networks, it is possible that a link between two nodes may not work equally well in both directions, due to differing antenna or propagation patterns or sources of interference [28].

DSR allows such unidirectional links to be used when necessary, improving overall performance and network connectivity in the system.

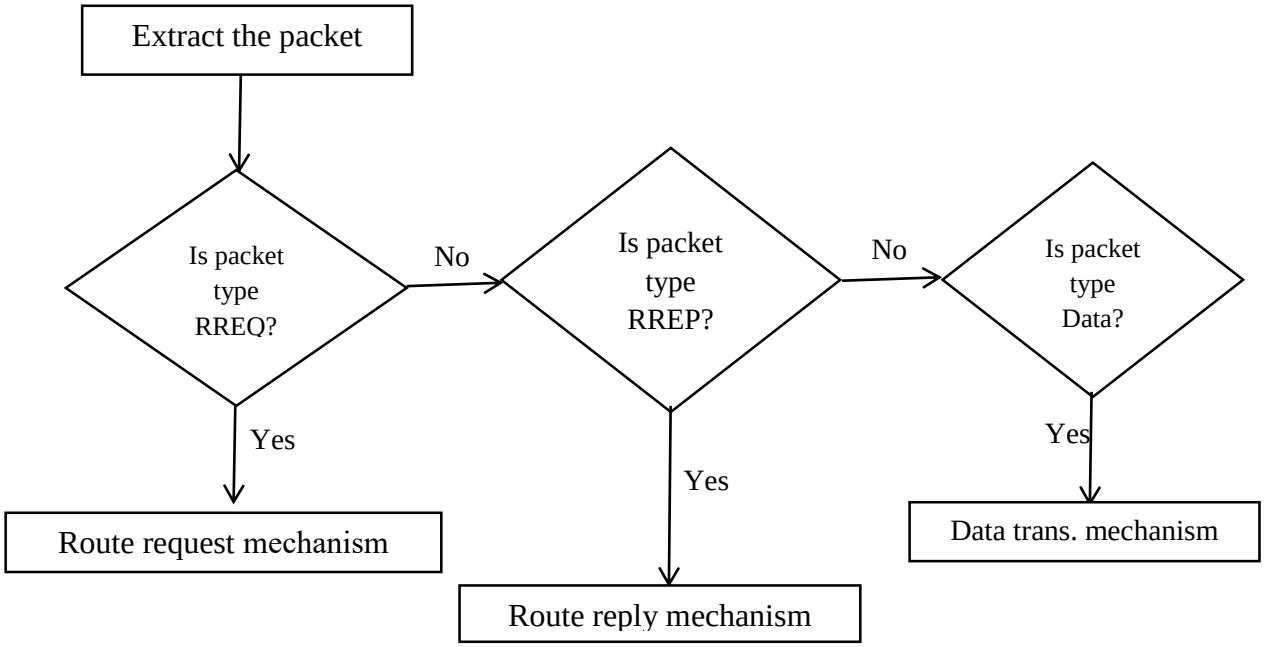


Figure 14: Flow chart for DSR routing mechanism [28]

DSR also supports internetworking between different types of wireless networks, allowing a source route to be composed of hops over a combination of any types of networks available. For example, some nodes in the Ad-hoc network may have only short-range radios, while other nodes have both short-range and long-range radios; the combination of these nodes together can be considered by DSR as a single Ad-hoc network. In addition, the routing of DSR has been integrated into standard Internet routing, where a “gateway” node connected to the Internet also participates in the Ad-hoc network routing protocols; and has been integrated into Mobile IP routing.

4.1.1 BASIC DSR ROUTE DISCOVERY

When some node **S** originates a new packet destined to some other node **D**, it places in the header of the packet a source route giving the sequence of hops that the packet should follow on its way to **D**. Normally will obtain a suitable source route by searching its Route Cache of routes previously learned, but if no route is found in its cache, it will initiate the Route Discovery protocol to dynamically find a new route to **D** [29].

In this case, we call **S** the initiator and **D** the target of the Route Discovery. For example, figure 10 illustrates an example Route Discovery, in which a node **A** is attempting to discover a route to node **E**. To initiate the Route Discovery, **A** transmits a ROUTE REQUEST message as a single local broadcast packet, which is received by (approximately) all nodes currently within wireless transmission range of **A**. Each ROUTE REQUEST message identifies the initiator and target of the Route Discovery, and contains a unique request id, determined by the initiator of the REQUEST. Each ROUTE REQUEST also contains a record listing the address of each intermediate node through which this particular copy of the ROUTE REQUEST message has been forwarded. This route record is initialized to an empty list by the initiator of the Route Discovery.

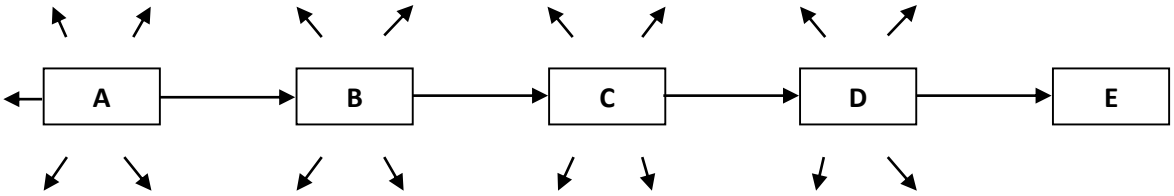


Figure 15: Route Discovery Example: **A** is the initiator and **E** is the target [29]

When another node receives a ROUTE REQUEST, if it is the target of the Route Discovery, it returns a ROUTE REPLY message to the initiator of the Route Discovery, giving a copy of the accumulated route record from the ROUTE REQUEST; when the initiator receives this ROUTE REPLY, it caches this route in its Route Cache for use in sending subsequent packets to this destination. Otherwise, if this node receiving the ROUTE REQUEST has recently seen another ROUTE REQUEST message from this initiator bearing this same request id, or if it finds that its own address is already listed in the route record in the ROUTE REQUEST message, it discards the REQUEST.

Otherwise, this node appends its own address to the route record in the ROUTE REQUEST message and propagates it by transmitting it as a local broadcast packet (with the same request id).

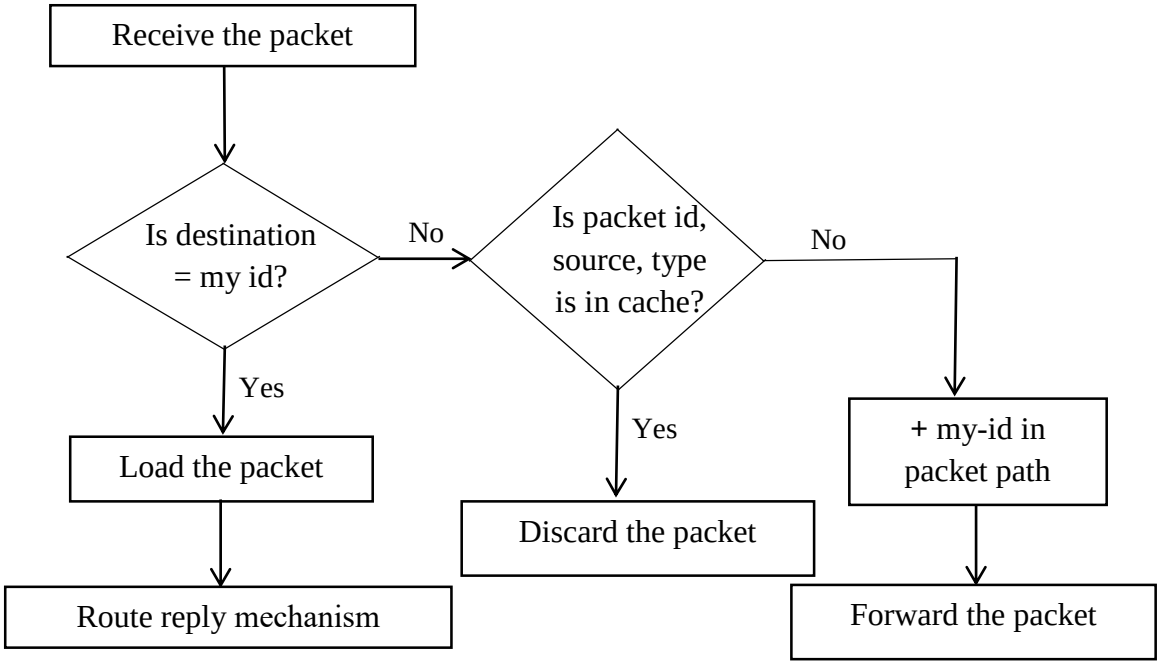


Figure 16: Flow chart for Route Request Mechanism [28]

In returning the ROUTE REPLY to the initiator of the Route Discovery, such as node **E** replying to **A** in figure 15, node **E** will typically examine its own Route Cache for a route back to **A**, and if found, will use it for the source route for delivery of the packet containing the ROUTE REPLY. Otherwise, **E** may perform its own Route Discovery for target node **A**, but to avoid possible infinite recursion of Route Discoveries, it must piggyback this ROUTE REPLY on its own ROUTE REQUEST message for **A**. It is also possible to piggyback other small data packets, such as a TCP SYN packet, on a ROUTE REQUEST using this same mechanism. Node **E** could also simply reverse the sequence of hops in the route record that it trying to send in the ROUTE REPLY, and use this as the source route on the packet carrying the ROUTE REPLY itself.

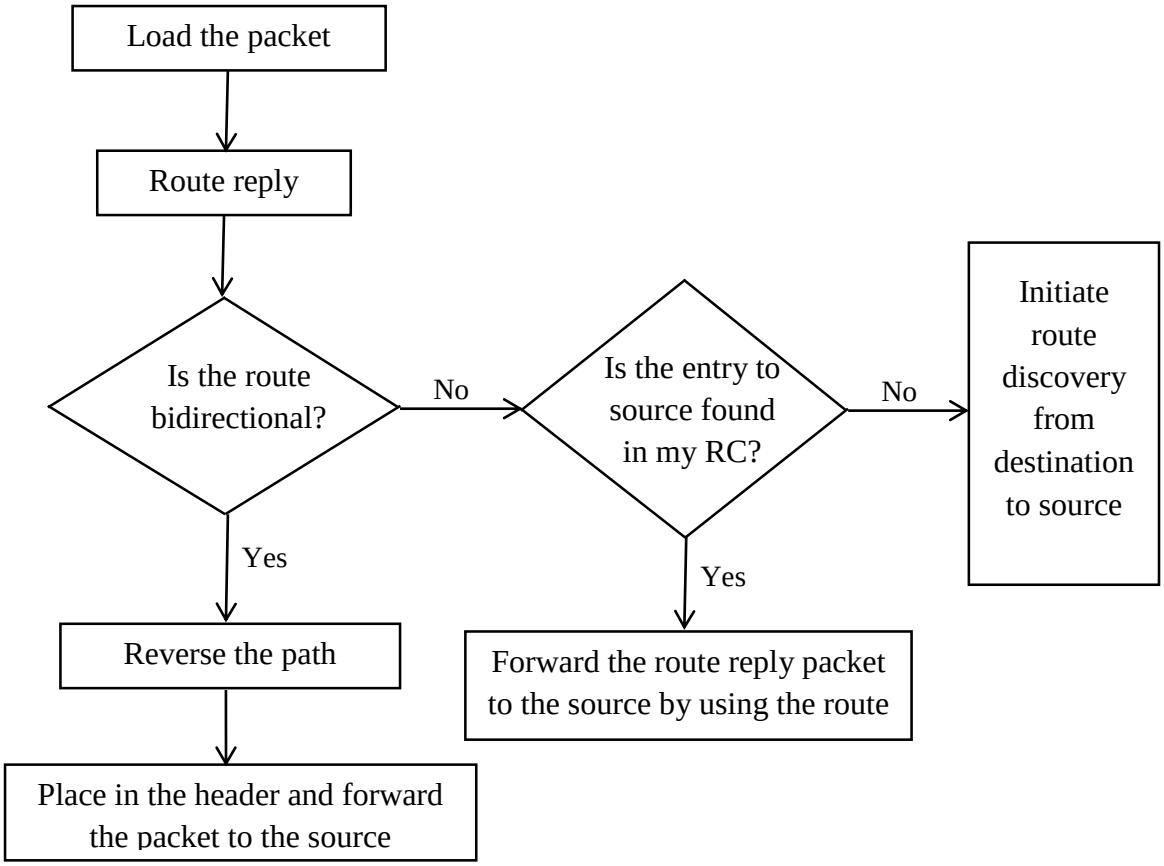


Figure 17: Flow chart for Route Reply Mechanism [28]

For MAC protocols, such as IEEE 802.11 that require a bi-directional frame exchange as part of the MAC protocol [IEEE 1997], this route reversal is preferred as it avoids the overhead of a possible second Route Discovery, and it tests the discovered route to ensure it is bi-directional before the Route Discovery initiator begins using the route. However, this technique will prevent the discovery of routes using unidirectional links. In wireless environments where the use of unidirectional links is permitted, such routes may in some cases be more efficient than those with only bi-directional links, or they may be the only way to achieve connectivity to the target node.

When initiating a Route Discovery, the sending node saves a copy of the original packet in a local buffer called the Send Buffer. The Send Buffer contains a copy of each packet that cannot be transmitted by this node because it does not yet have a source route to the packet's destination. Each packet in the Send Buffer is stamped with the time that it was placed into the Buffer and is discarded after residing in the Send Buffer for some timeout period; if necessary for preventing the Send Buffer from overflowing, a First-In First-Out (FIFO) or other replacement strategy can also be used to evict packets before they expire. While a packet

remains in the Send Buffer, the node should occasionally initiate a new Route Discovery for the packet's destination address. However, the node must limit the rate at which such new Route Discoveries for the same address are initiated, since it is possible that the destination node is not currently reachable. Due to the limited wireless transmission range and the movement of the nodes in the network, the network may at times become partitioned, meaning that there is currently no sequence of nodes through which a packet could be forwarded to reach the destination. Depending on the movement pattern and the density of nodes in the network, such network partitions may be rare or may be common [29].

If a new Route Discovery was initiated for each packet sent by a node in such a situation, many unproductive ROUTE REQUEST packets would be propagated throughout the subset of the Ad-hoc network reachable from this node. To reduce the overhead from such Route Discoveries, we use exponential back-off to limit the rate at which new Route Discoveries may be initiated by any node for the same target.

If the node attempts to send additional data packets to this same node more frequently than this limit, the subsequent packets should be buffered in the Send Buffer until a ROUTE REPLY is received, but the node must not initiate a new Route Discovery until the minimum allowable interval between new Route Discoveries for this target has been reached. This limitation on the maximum rate of Route Discoveries for the same target is like the mechanism required by Internet nodes to limit the rate at which ARP REQUESTs are sent for any single target IP address.

4.1.2 BASIC DSR ROUTE MAINTENANCE

When originating, or forwarding a packet using a source route [14], each node transmitting the packet is responsible for confirming that the packet has been received by the next hop along the source route; the packet is retransmitted (up to a maximum number of attempts) until this confirmation of receipt is received. For example, in the situation illustrated in figure 13, node **A** has originated a packet for **E** using a source route through intermediate nodes **B**, **C**, and **D**. In this case, node **A** is responsible for receipt of the packet at **B**, node **B** is responsible for receipt at **C**, node **C** is responsible for receipt at **D**, and node **D** is responsible for receipt finally at the destination **E** [29].

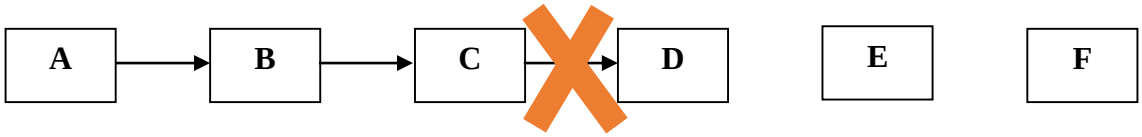


Figure 18: Route Maintenance: Node **C** is unable to forward a packet to next hope node **D** [35]

This confirmation of receipt in many cases may be provided at no cost to DSR, either as an existing standard part of the MAC protocol in use (such as the link-level acknowledgement frame defined by IEEE 802.11 [IEEE 1997]), or by a passive acknowledgement (in which, for example, **B** confirms receipt at **C** by overhearing **C** transmit the packet to forward it on to **D**). If neither of these confirmation mechanisms are available, the node transmitting the packet may set a bit in the packet's header to request a DSR-specific software acknowledgement be returned by the next hop; this software acknowledgement will normally be transmitted directly to the sending node, but if the link between these two nodes is unidirectional, this software acknowledgement may travel over a different, multi-hop path.

If the packet is retransmitted by some hop the maximum number of times and no receipt confirmation is received, this node returns a ROUTE ERROR message to the original sender of the packet, identifying the link over which the packet could not be forwarded. For example, in figure 18, if **C** is unable to deliver the packet to the next hop **D**, then **C** returns a ROUTE ERROR to **A**, stating that the link from **C** to **D** is currently "broken." Node **A** then removes this broken link from its cache; any retransmission of the original packet is a function for upper layer protocols such as TCP. For sending such a retransmission or other packets to this same destination **E**, if **A** has in its Route Cache another route to **E** (for example, from additional ROUTE REPLYs from its earlier Route Discovery, or from having overheard sufficient routing information from other packets), it can send the packet using the new route immediately. Otherwise, it may perform a new Route Discovery for this target [29].

4.2 COOPERATIVE IMPLEMENTATION IN DSR PROTOCOL

In proposed cooperative retransmission, the cooperation will take place in MAC layer to minimize the delay and traffic bottleneck but when the direct transmission failed, before sending transmission failure to upper layer (network layer) feedback support cooperative enabled message will be initiated.

The following commands take the message from Data link layer;

```
Void CoopEnableCallback (Packet *pkt, void *data)
```

```
{
  DSRAgent *agent = (DSRAgent *)data;
  agent->coopEnabled(pkt);
}
```

From this callback the following coopEnabled function will called;

```
Void DSRAgent::coopEnabled(Packet *pkt, const char* reason)
```

CHAPTER FIVE

5. RESULTS

5.1 BACKGROUND ON NS-2 SIMULATOR

NS is a discrete event simulator, where the advance of time depends on the timing of events which are maintained by the scheduler. An event is an object in C++ hierarchy with a unique ID, a scheduled time and the pointer to an object that handles the event. The scheduler keeps an ordered data structure with the events to be executed and fires them one by one invoking the header of the event [30].

NS is based on two languages: an object-oriented simulator, written in C++, and an OTCL (Object oriented extension of Tool Command Language) interpreter, used to execute users' command scripts [35].

NS has a rich library of network and protocol objects. There are two class hierarchies: the compiled C++ hierarchy and the interpreted OTCL one, with one to one correspondence between them. The compiled C++ hierarchy allows us to achieve efficiency in the simulation and faster execution times. This is useful for the detailed definition and operation of protocols. This allows one to reduce packet and event processing time.

When performing a network simulation, users first write an OTCL script to describe the relevant network topology, and to define traffic sources and when to start and stop transmitting packets through an event scheduler. NS-2 has an OTCL interpreter that translates the event scheduler and the network component OTCL objects and member functions into their corresponding C++ counterparts. Such a mechanism provides a linkage between the OTCL and C++ realms, making the control functions and the configurable variables interoperable. Next, NS-2's main program simulates the topology with user-specified parameters.

Eventually, the simulation is finished; NS-2 generates an output trace file that contains detailed simulation data [30]. The data can be either post-processed to create simulation graphs or input to a graphical simulation tool called Network Animator (NAM) for later viewing. The architecture of NS-2 is shown in the figure below.

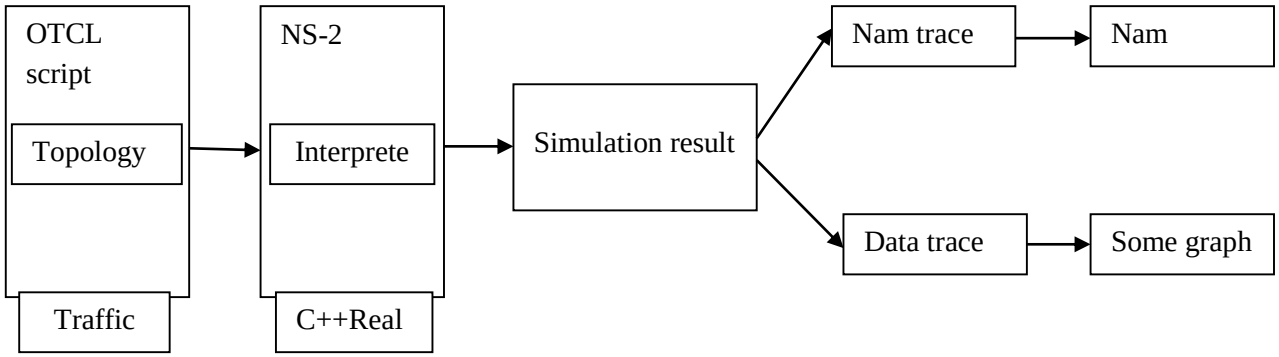


Figure 19: Overall architecture of NS-2 [32]

After the simulation finished, the way NS-2 used to present the most details information on that much network layer is that it provides us a huge trace file recording all the events line by line in it. So, now we see why event driven mechanism is used in NS-2, since it really could maintain the things ever happened as records. And we can trace these records to evaluate the performance of special stuffs in our network, such as routing protocol, Mac layer load, and so on [19].

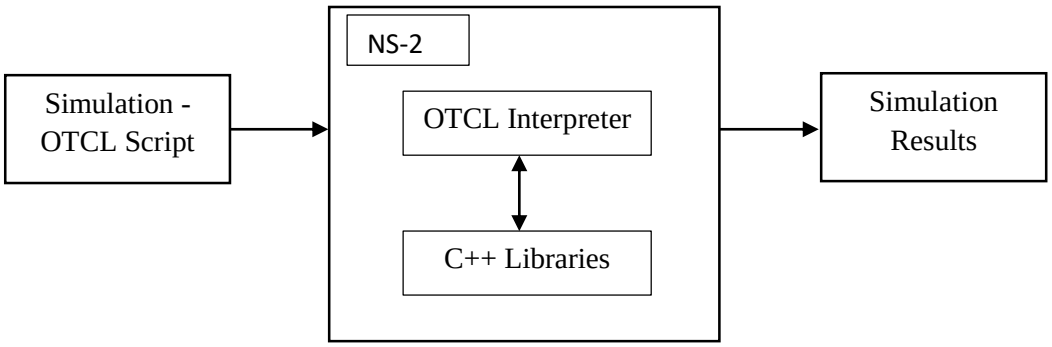


Figure 20: Data flow of one time simulation in NS-2 [33]

As figure 20 shows, for the data flow of one time simulation in NS-2, the user input an OTcl source file, the OTcl script do the work of initiating an event scheduler, sets up the network topology using the network objects and the plumbing functions in the library, and tells traffic sources when to start and stop transmitting packets through the event scheduler. And then, this OTcl script file will be passed to NS-2, in this view, we can treat NS-2 as Object-oriented Tcl (OTcl) script interpreter that has a simulation event scheduler and network component object libraries, and network setup module libraries. And then the detail network construction and traffic simulation will be done in NS-2. After a simulation is finished, NS-2 produces one or more text-based output files that contain detailed simulation data, and that data can be used for simulation analysis [30].

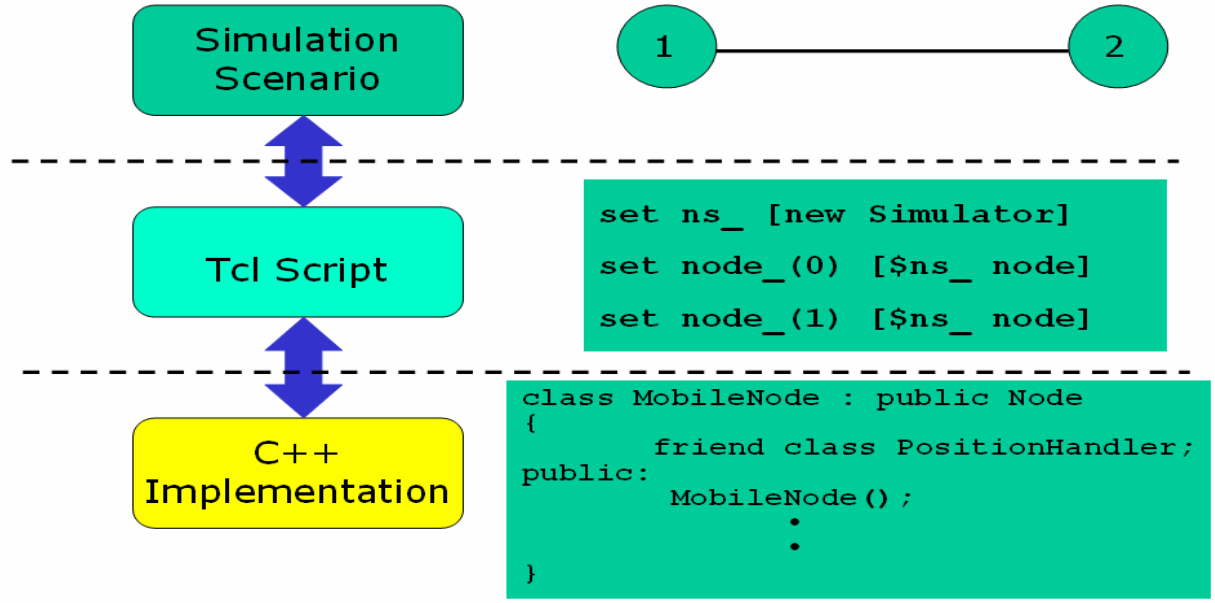


Figure 21: Layered architecture of NS-2 [33]

Figure 21 shows the layered architecture of NS-2. The event schedulers and most of the network components are implemented in C++ and are available to Tcl Script, thus the lowest level of NS-2 is implemented by C++, and the Tcl script level is on top of it to make simulation stuffs much easier to be conducted. Then, upon the Tcl level, we see the overview of the network. That is the simulation scenario. These all things combined as so called NS-2 software [34].

5.2 PARTS NEEDED BY ONE SIMULATION IN NS-2

To successfully carry out one simulation, we must first tell NS-2 things it may need from us for one simulation. So, what we need is to follow three necessary items [30, 34]:

1) **Appearance of the network:** the whole topology view of sensor network or mobile network, this includes the position of nodes with (x, y, z) coordinate, the node movement parameters, the movement starting time, the movement is to what direction, and the node movement speed with pausing time between two supposed movement.
2) **Internal of the network:** Since the simulation is on the network traffic, so it is important we tell the NS-2 about which nodes are the sources, how about the connections, what kind of connection we want to use.
3) **Configuration of the layered structure** of each node in the network, this includes the detail configuration of network components on mobile node or sensor node, and also we need to drive the simulation, so we need to give out where to give out the simulation results which is the trace file, and how to organize a simulation process.

5.3 SIMULATION ANALYSIS OF COOPERATIVE RETRANSMISSION

The intent of this section is to evaluate the effectiveness of proposed cooperative implementation in DSR with respect to the presently existing mechanism in Dynamic Source Routing protocol.

To evaluate the two mechanisms, different parameters have been used, keeping the following parameters constant.

- Simulation coverage area i.e. 1000x600m

Data rate is the amount of data transfer within a certain time.

- Seed value of 1.0

A seed value of 1.0 is used to obtain the same value of random variable while running the simulator again and again.

- CBR traffic generation

CBR is a Constant Bit Rate with an application over UDP connection.

The basic parameters that have been used in this work for performance evaluations includes:

I. THROUGHPUT

The throughput (messages/second) is the total number of delivered data packets divided by the total duration of simulation time. Evaluation of throughput enables us to measure how efficient the mechanisms are under different scenarios.

II. PDF

PDF is Packet Delivery Fraction (Ratio). Packet delivery ratio is calculated by dividing the number of packets received by the destination through the number of packets originated by the application layer of the source (i.e. Constant Bit Rate (CBR)). It specifies the packet loss rate, which limits the maximum throughput of the network. The better the delivery ratio, the more complete and correct is the routing protocol.

III. PAUSE TIME

Pause time is the time that nodes within the wireless communication region are stop moving. It is usually in seconds in Ad-hoc network. Different values of pause time are generated to evaluate its effect against parameters like PDF and throughput.

IV. OVERHEAD

Overhead is the exchange of packets within the network. In Ad-hoc network there is a continuous exchange of data packets, acknowledgment packet, data request packet and other packets. Hence, this work evaluated how efficient cooperative retransmission is in avoiding excessive overhead that could be created within an Ad-hoc network.

V. AVERAGE END-TO-END DELAY

Average End-to-End delay (in seconds) is the average time it takes a data packet to reach the destination. This metric is calculated by subtracting "time at which first packet was transmitted by source" from "time at which first data packet arrived to destination". This includes all possible delays caused by buffering during route discovery latency, queuing at the interface queue, retransmission delays at the MAC, propagation and transfer times. This metric is significant in understanding the delay introduced by path discovery.

SIMULATOR INTERFACE

Network simulator NS-2.35 is used in simulating an Ad-hoc network scenario with different moving nodes and pause times. The following figure shows the simulation environment of NS-2.35 with 16 dynamic nodes. Note that the circular regions in the simulator shows node discovery regions or node coverage area.

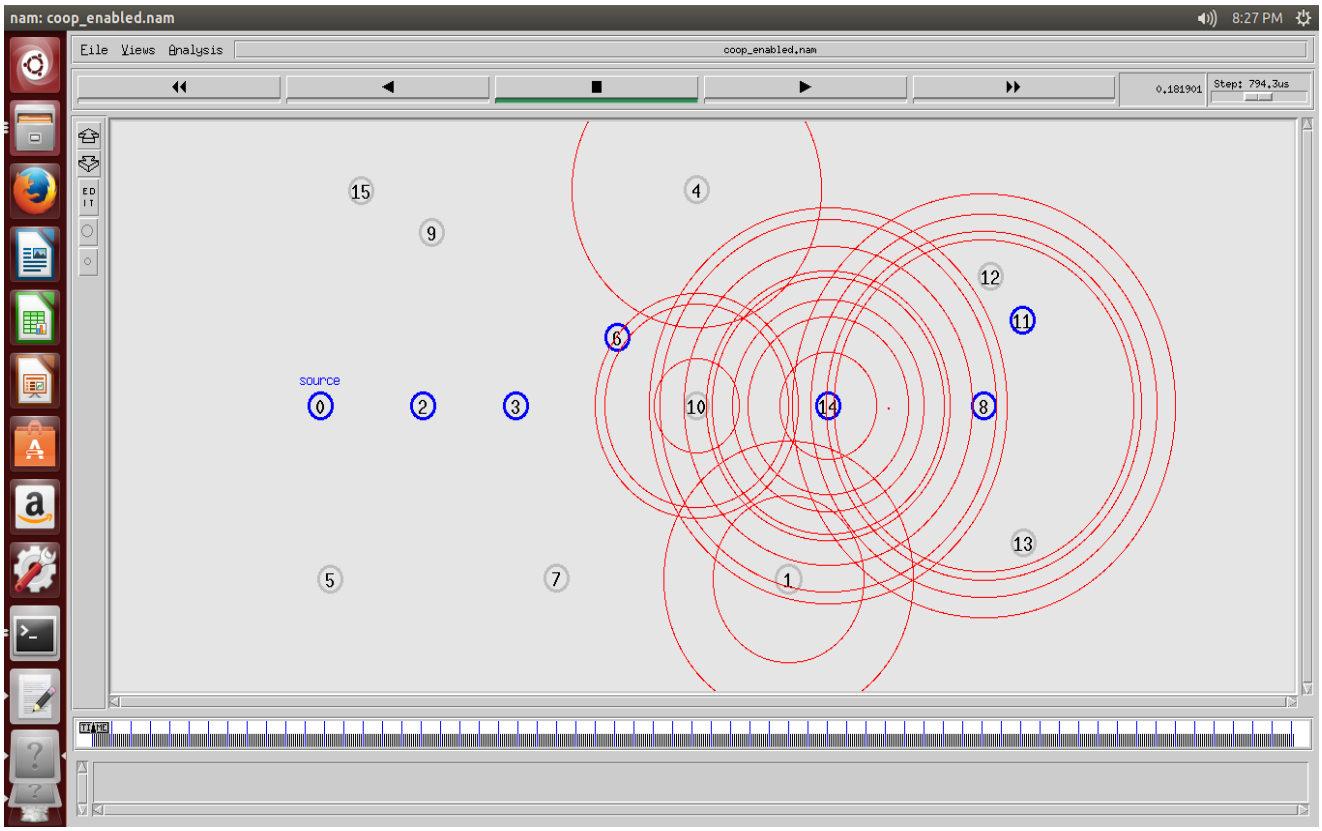


Figure 22: NS-2.35 interface

5.3.1 PERFORMANCE EVALUATION OF COOPERATIVE RETRANSMISSION

Performance evaluation and comparison is grouped in to two broad categories based on the number of nodes involved and pause times, throughout the simulation time.

In the first part of performance evaluation, comparison made between Coop-Adhoc and the presently existing Ad-hoc by keeping pause time and node speed constant while varying the number of nodes over the range of 5 to 55 nodes.

However, in the second parts of performance evaluation, number of moving nodes fixed to 35 and pause time is varied from 1 to 5 seconds.

Notice: in simulation result analysis

- **Line -** represents **Coop-Adhoc** and
- **Error Line -** represents **Ad-hoc**

FIRST PART OF SIMULATION RESULT ANALYSIS

In this study, the used performance evaluation parameters are throughput, packet delivery fraction, normalized routing load and end to end delay, against number of nodes ranging from 5 to 55, by keeping the following parameters constant.

- ✓ Pause time = 1sec
- ✓ Maximum connection = 5
- ✓ Minimum speed of nodes = 10m/s
- ✓ Maximum speed of nodes =30m/s
- ✓ Coverage area of simulation = 1000mX600m
- ✓ Traffic type = CBR

THE EFFECT OF NUMBER OF NODES IN COOP-ADHOC AND AD-HOC COMPARISON:

a. THROUGHPUT

Throughput is the total number of delivered data packets divided by the total simulation duration and it enables us to measure efficiency of the mechanisms under different scenarios.

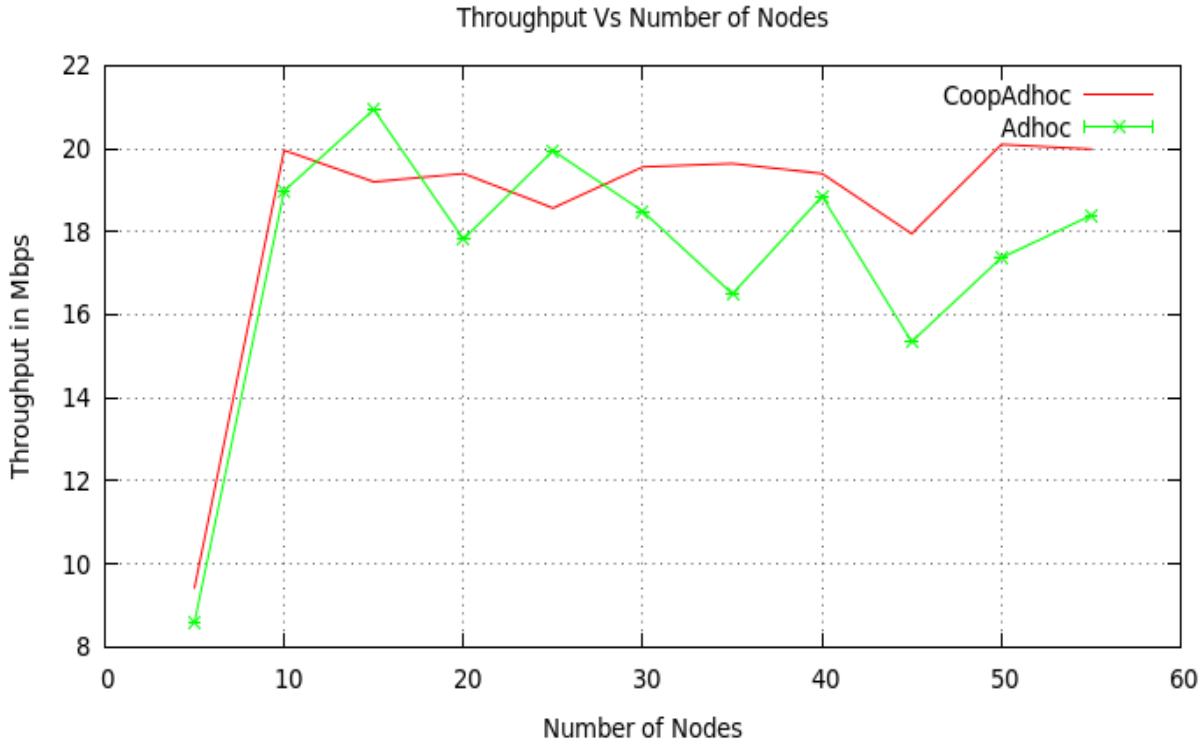


Figure 23: Throughput Vs Number of Nodes

b. PACKET DELIVERY FRACTION

PDF is the ratio of number of packets received by the destination and number of CBR packets originated by the application layer of the source.

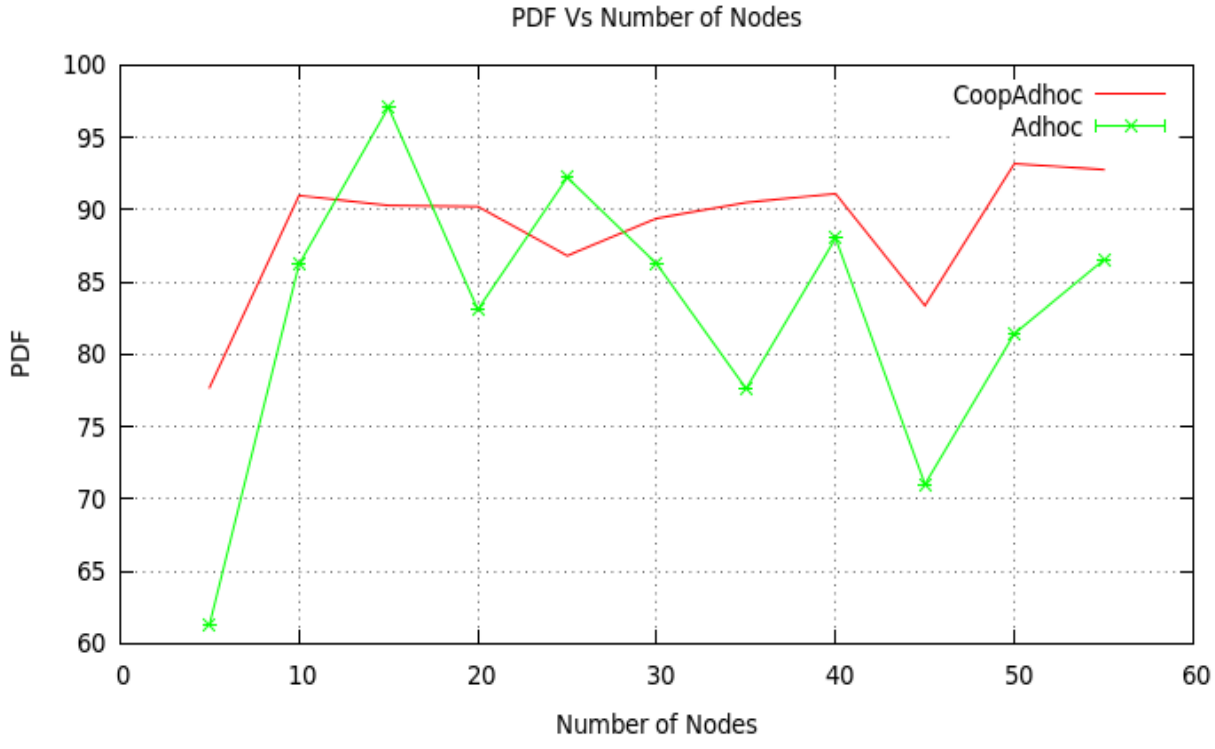


Figure 24: PDF Vs Number of Node

c. NORMALIZED ROUTING LOAD

The number of routing packets transmitted per data packet delivered at the destination. Each hop-wise transmission of a routing packet is counted as one transmission.

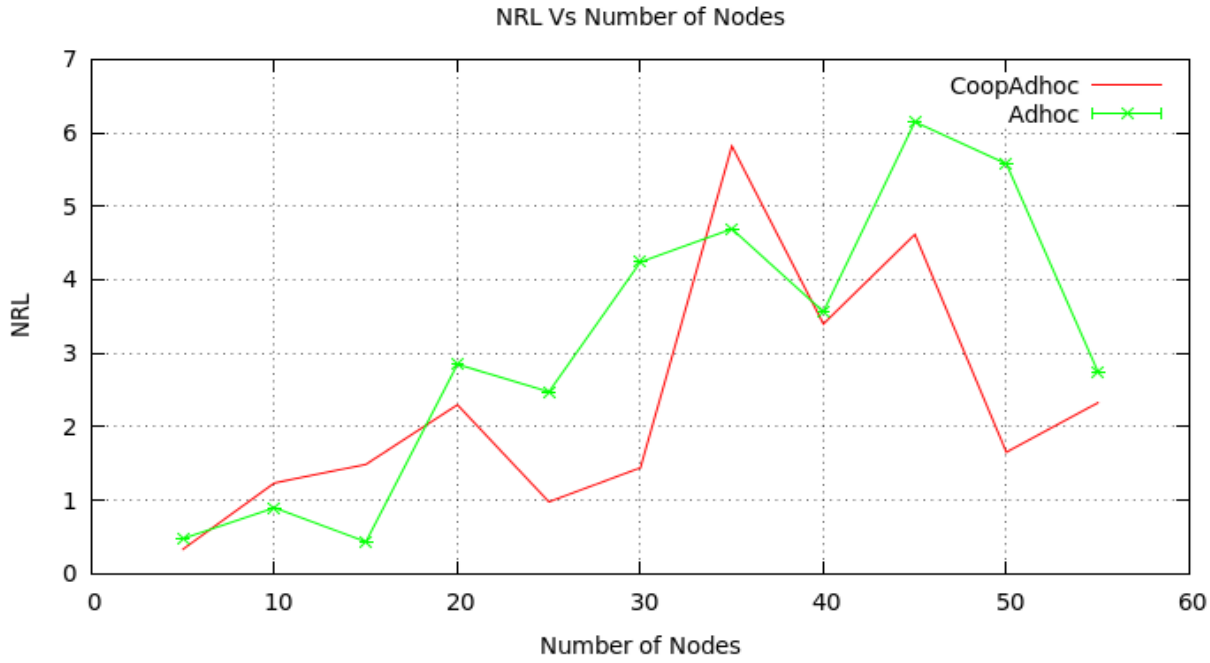


Figure 25: NRL Vs Number of Node

d. END TO END DELAY

End to end delay is the average time taken for a data packet to reach the destination and it is the difference between time at which first data packet arrived to destination and time at which first packet was transmitted by source.

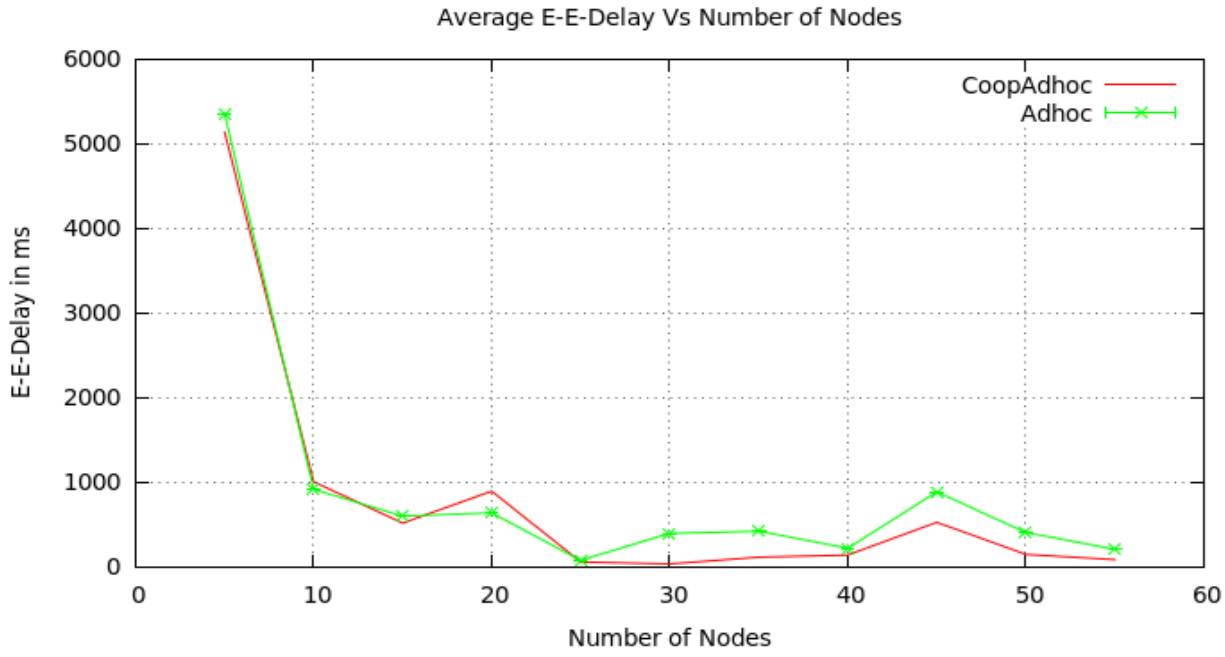


Figure 26: E-E-Delay Vs Number of Node

SECOND PART OF SIMULATION RESULT ANALYSIS

Performance parameters under study for result analysis are the same with those stated above, but against pause time 1,2,3,4 and 5 seconds, keeping the following parameters constant.

- Number of nodes = 35
- Maximum connection = 5
- Speed of nodes: Min = 10 m/s, Max = 30 m/s
- Coverage area of simulation = 1000mX600m
- Traffic type = CBR

THE EFFECT OF PAUSE TIME IN COOP-ADHOC AND AD-HOC:

a. THROUGHPUT

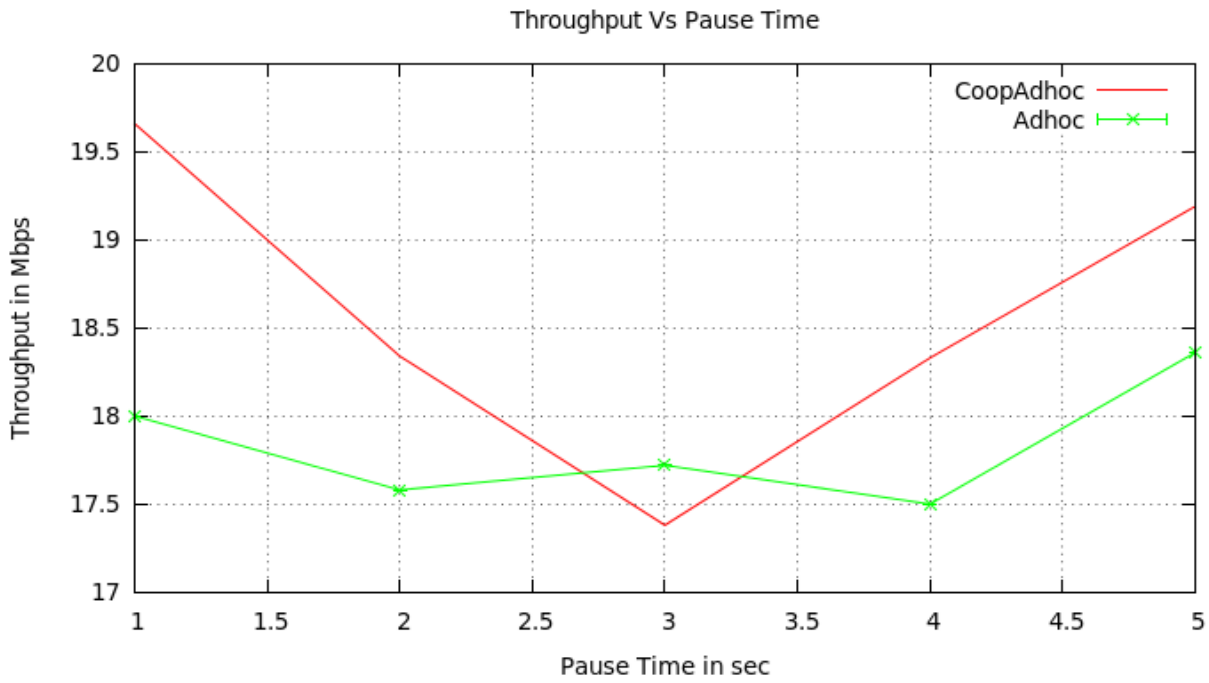


Figure 27: Throughput Vs Pause Time

b. PACKET DELIVERY FRACTION

PDF in one of performance measuring parameters, it could be calculated by dividing the number of packets received by the destination through the number of packets originated by the application layer of the source i.e. CBR.

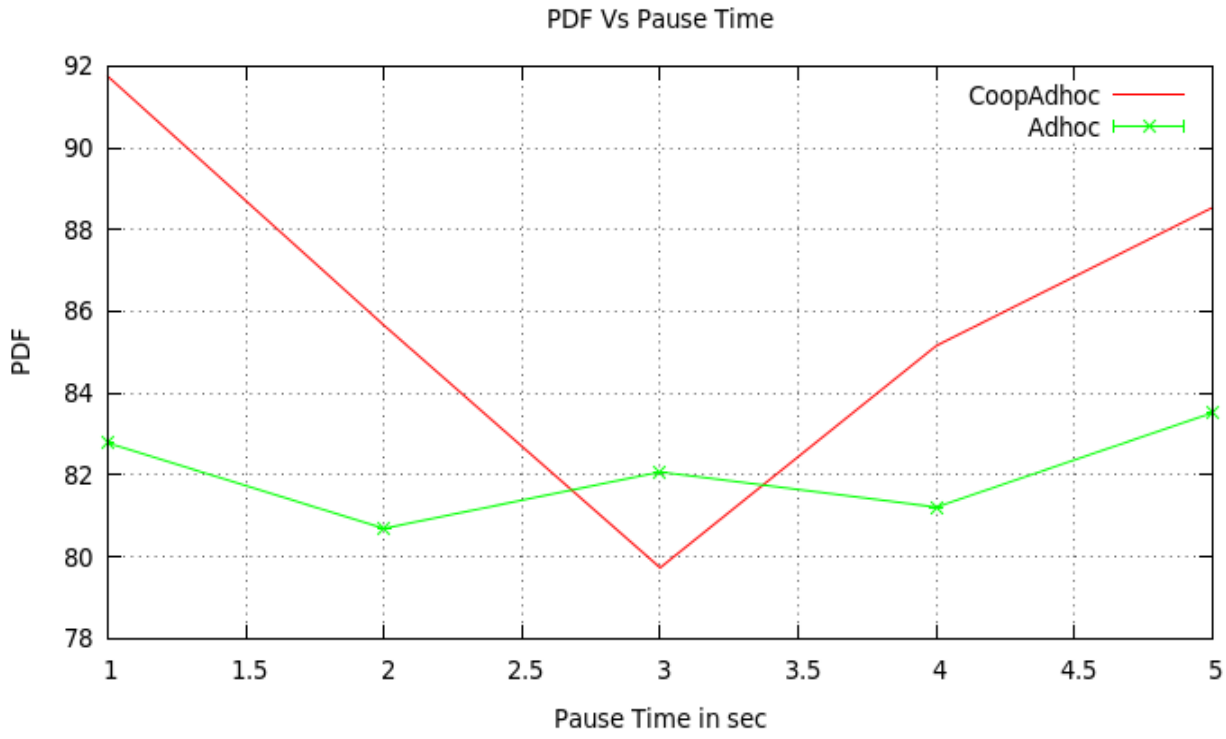


Figure 28: PDF Vs Pause Time

c. NORMALIZED ROUTING LOAD

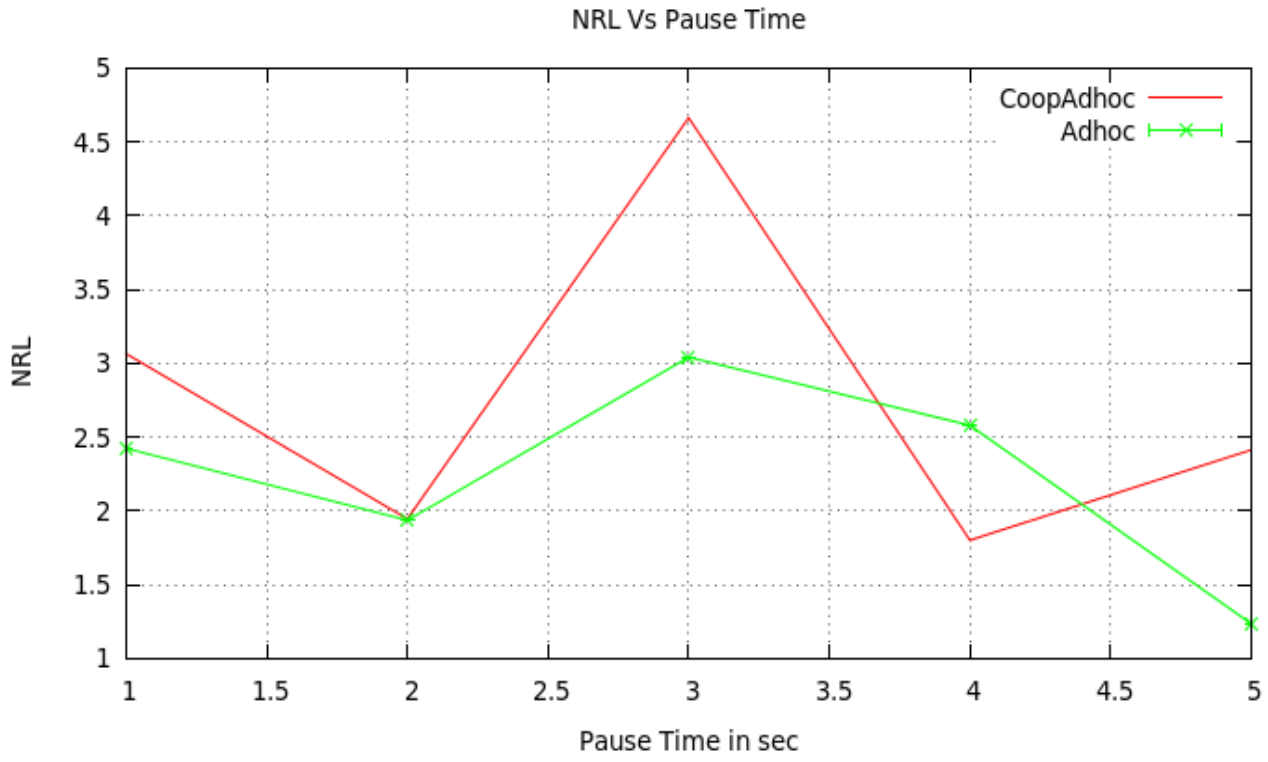


Figure 29: NRL Vs Pause Time

d. END TO END DELAY

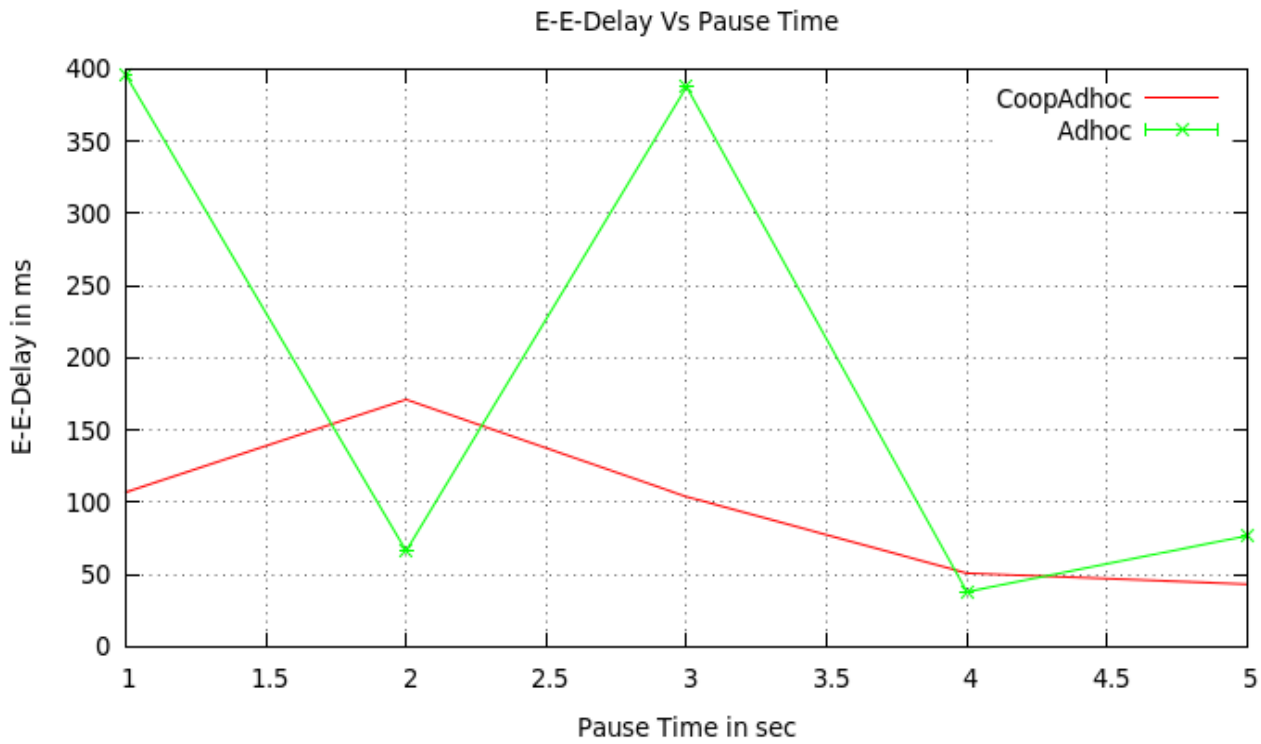


Figure 30: End to End Delay Vs Pause Time

RESULT ANALYSIS

This study shows that Cooperative communication implementation in Ad-hoc network using dynamic source routing (DSR) protocol have been enhanced some features of the Ad-hoc communication network by means of utilizing its features of MAC layer retransmission capabilities. The study results can be summarized as follows.

i. THROUGHPUT

As can be seen from figure 23, the throughput values of cooperative communication are higher than that of Ad-hoc communication specifically for number of nodes of less than 12 and for that of more than 28 nodes.

Similarly, from figure 27, the cooperative Ad-hoc outperform the legendary Ad-hoc for all points of pause times except at 3 seconds pause time.

ii. PACKET DELIVERY FRACTION (PDF)

Cooperative communications main target is to accomplish the ongoing communication that is broken either due to battery outage or coverage area problem. PDF is ratio of the number of packets received by the destination node to the number of packets originated by the application layer of the source node. The cooperative retransmission main feature is to maintain the broken connection by cooperation request frames, thus the packet delivery fraction of the cooperative Ad-hoc is far greater than that of Ad-hoc communication. As can be seen from figure 24 and figure 28 for number of nodes and pause time respectively, Coop-Adhoc performs better than standard Ad-hoc communication.

iii. NORMALIZED ROUTING LOAD (NRL)

Number of routing packets "transmitted" per data packet "delivered" at destination. Each hop-wise transmission of a routing is counted as one transmission. It is the sum of all control packet sent by all node in network to discover and maintain routes. As shown on figure 25 Coop-Adhoc performs better than Ad-hoc for number of nodes between 18 and 33, and when the number of mobile nodes are more than 40. However, in case of figure 29 that is NRL versus pause time, the Ad-hoc performs better than Coop-Adhoc except for the interval of 3.7 and 4.5 seconds pause time.

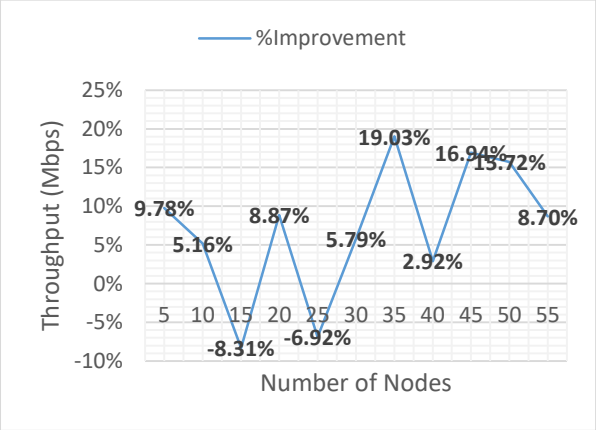
iv. END TO END DELAY

End to end delay includes all possible delay caused by buffering during route discovery latency, queuing at the interface queue, retransmission delay at the MAC, propagation and transfer time. Generally, it is the time taken for a data packet to be transmitted across an MANET from source to destination. Average end-to-end delay of Coop-Adhoc is comparable to Ad-hoc for number of nodes less than 25. But, with the increase in network load (greater than 25 nodes), the average delay of Ad-hoc is higher than Coop-Adhoc as shown in figure 26. Similarly, in case of pause time (figure 30) the average delay of Coop-Adhoc is less than that of Ad-hoc in general except for pause time (between 1.7 and 2.3 sec).

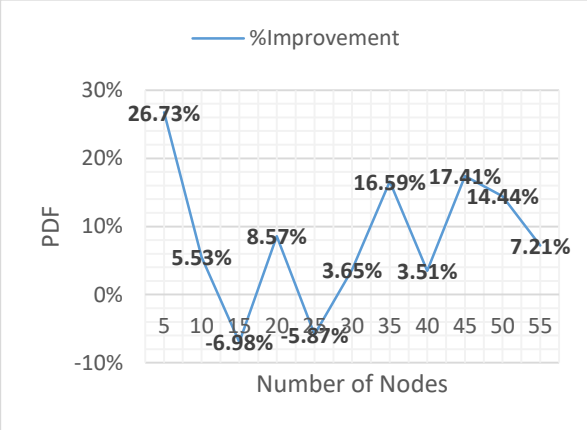
From simulation results, this study shows that Cooperative communication in mobile Ad-hoc network (MANET) using DSR protocol is recommended for number of nodes and pause times specified above. This is because in Ad-hoc networks when network load is increased, route failure maintenance at MAC layer is difficult, hence sending failure message from the node

next to broken link to the originating source could increase delay. In addition to that, the source node must initiate the route discovery to the destination node once again. But in case of Coop-Adhoc the node next to broken link will act as an intermediate virtual source and initiate the cooperation in network layer level rather than sending the link failure message to the source node. So again, logically when network loads increase the Coop-Adhoc would perform better than Ad-hoc. In general, as cooperative communication in MANET is designed to maintain link breakage with minimum overhead, the simulation result goes with the principle.

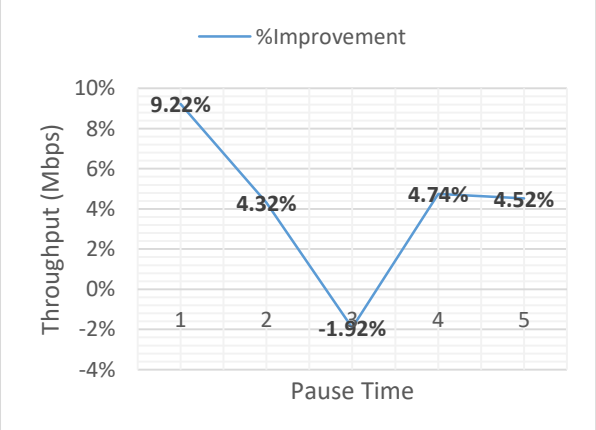
In addition to this, cooperative communication performs well for high number of nodes since the distance between source and destination nodes may increase. This in turn increases the number of intermediate nodes participating in routing a packet. If in case link breakage happen, Coop-Adhoc maintains it through initializing route discovery by sending RFC message to the nodes found in 1 hop ranges to cooperate to send the data to the required destination, rather than sending the route failure message back to the source.



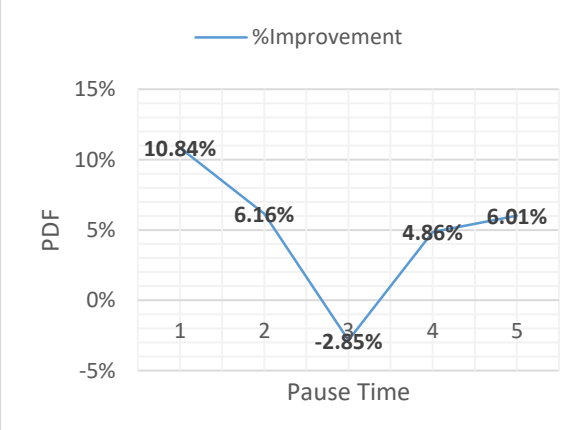
(a)



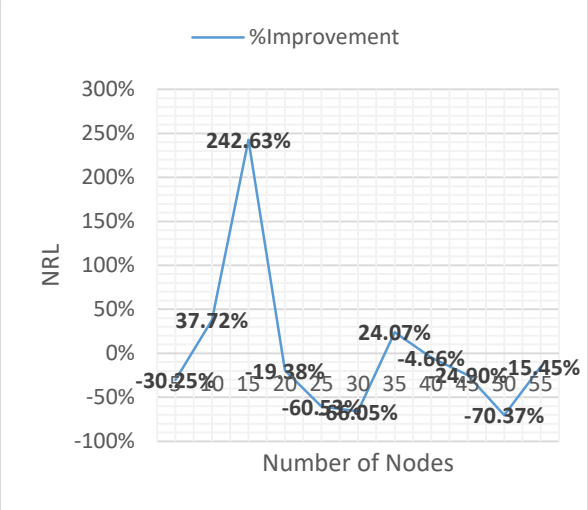
(c)



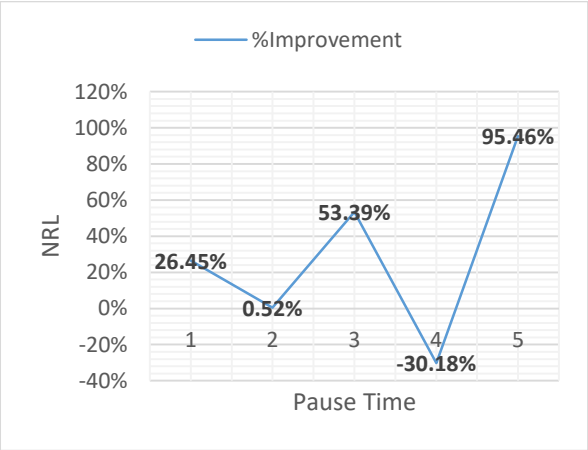
(b)



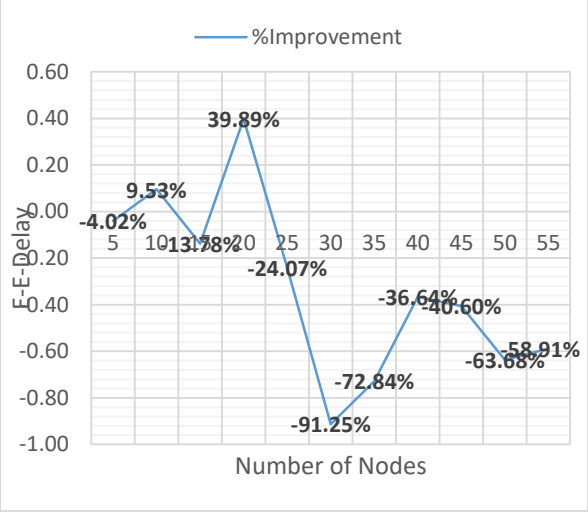
(d)



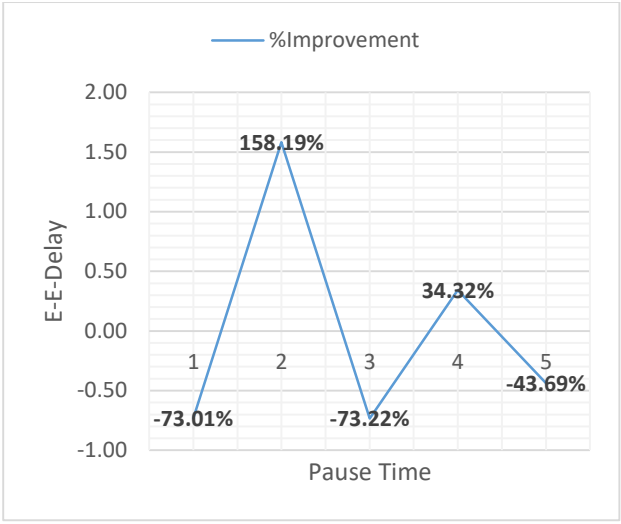
(e)



(f)



(g)



(h)

Figure 31: Percentage Change Comparison

In overall analysis of performance comparison, the Coop-Adhoc throughput improved by 7.1% and 4.2% versus number of nodes and pause times respectively. Similarly, Packet delivery fraction of Coop-Adhoc improved over Adhoc by 8.3% for different number of nodes and by 5% against different pause time. Significant improvement which is 32.4% is registered for end to end delay versus number of nodes.

Since main target of cooperative communication is delivering the packet sent to the destination with less delay and routing loads, the result of this work is aligned with this principle, and outperformed non-cooperative Adhoc and improved the normalized routing load against pause time by 29%.

CHAPTER SIX

6. CONCLUSION AND RECOMMENDATION

6.1 CONCLUSION

Since the research approach of route maintenance is based on the presently existing MAC layer and DSR, the study cover the basic features of DSR mainly and other routing protocols for comparison purpose. As well as, the interdependence between MAC layer and network layer.

As stated, this study emphasized on finding alternative route maintenance mechanism for DSR to enhance its contemporary efficiency on metrics like throughput, packet delivery ratio, normalized routing load and average end to end delay. To examine these, simulation based performance evaluations are used for Coop-Adhoc and Ad-hoc, by varying pause time and number of nodes against those performance metric parameters.

The studied results show that; the throughput values of cooperative communication are higher than that of Ad-hoc communication specifically for number of nodes of less than 12 and for that of more than 28 nodes. Similarly, for all points of pause times except at 3 seconds pause time.

The packet delivery fraction of the cooperative Ad-hoc is far greater than that of Ad-hoc communication. As can be seen from figure 24 for number of nodes and figure 28 for pause time.

In case of normalized routing load again Coop-Adhoc performs better than Ad-hoc for number of nodes between 18 and 33, and when the number for mobile nodes are more than 40.

When network load is more than 25 nodes, average end to end delay of Coop-Adhoc is lower than that of Ad-hoc. Similarly, in case of pause time the average delay of Coop-Adhoc is less than that of Ad-hoc in general except for pause time (between 1.7 and 2.3 sec).

This study could be enhanced by analyzing MANET routing protocols other than DSR protocol, under different mobility models and traffic sources, with respect to those performance metrics used here and others. Specially, by investigating further on cross layer optimization.

6.2 RECOMMENDATION AND FUTURE WORK

Simulation results of this study show that, Coop-Adhoc performs better than Ad-hoc for the metrics that are considered under those specific scenarios. If these scenarios meet with real world phenomena, the improved protocol would be chosen over nominal Ad-hoc. As a future work, additional adjustment like cross layer optimization should be considered for this protocol, to achieve better performance.

In the network simulator core program, it is impossible to independently modify and evaluate one layer, since all the layers should be adjusted to understand each other and to have effective communication across the layer. Thus, cross layer optimization should be considered for better and concrete performance level.

APPENDIX

The following commands take the message from Data link layer;

```
Void CoopEnableCallback (Packet *pkt, void *data)
{
 DSRAgent *agent = (DSRAgent *)data;
 agent->coopEnabled(pkt);
}
```

From this callback the following coopEnabled function will called;

```
Void DSRAgent::coopEnabled(Packet *pkt, const char* reason)
 /* mark our route cache understand the failure of the link between
   srh[cur_addr] and srh[next_addr], and then create a route request
   message and initiate cooperative communication to send to the
   destination node with help of Helper/Relay node*/
{
 hdr_sr *srh = hdr_sr::access(pkt);
 hdr_ip *iph = hdr_ip::access(pkt);
 hdr_cmn *cmh = hdr_cmn::access(pkt);

 assert(cmh->size() >= 0);

 srh->cur_addr() -= 1;          // correct for inc already done on sending

 ID initiator_id(srh->addrs()[0].addr, (ID_Type) srh->addrs()
     [srh->cur_addr()].addr_type);
 ID coop_id(srh->addrs()[srh->cur_addr()].addr,
            (ID_Type) srh->addrs()[srh->cur_addr()].addr_type);
 ID target_id(srh->addrs()[srh->num_addrs()].addr,
            (ID_Type) srh->addrs()[srh->cur_addr()].addr_type);
 assert(coop_id == net_id || coop_id == MAC_id);
 srh->route_error() = 0;
 srh->route_reply() = 0;
```

```
  srh->route_request() = 1;
  srh->coop_request () = 1;
  srh->flow_header() = 0;
  srh->flow_timeout() = 0;

  SRPacket p(pkt, srh);
  p.dest = target_id;
  p.src = net_id;

  iph->daddr() = Address::instance().create_ipaddr(target_id.addr,RT_PORT);
  iph->dport() = RT_PORT;
  iph->saddr() = Address::instance().create_ipaddr(net_id.addr,RT_PORT);
  iph->sport() = RT_PORT;
  iph->ttl() = 1;
  cmh->ptype() = PT_DSR;              // cut off data
  //cmh->size() = IP_HDR_LEN;
  cmh->size() = size_ + IP_HDR_LEN; // add in IP header
  //cmh->num_forwards() = 0;
  // assign this packet a new uid, since we're sending it
  cmh->uid() = uidcnt_++;

  /* send out route request */
  sendOutPacketWithRoute(p, true);
}
```

REFERENCES

[1] J. Loo, J. Mauri and J. Ortiz. *Mobile Ad Hoc Networks: Current Status and Future Trends.* Parkway NW, CRC Press, 2012, pp. 3-20.

[2] R. Hamamreh and M. Bawatna. "Protocol for Dynamic Avoiding End-to-End Congestion in MANETs." *Journal of Wireless Networking and Communications*, vol. 4(3), pp. 67-75, Apr. 2014.

[3] J. Jubin and J. Tornow. "The DARPA packet radio network protocols," in *Proc. of the IEEE*, vol. 75(1), 1987, pp. 21.32.

[4] F. Erbas, K. Kyamakya and K. Jobmann "Use of Position Information of Nodes in Mobile Ad hocNetworks," in *Proc. WPNC*, 2004, pp. 49-54.

[5] L. Bao and J. Garcia. "Topology Management in Ad Hoc Networks," in *Proc. MobiHoc*, 2003, pp. 129-140.

[6] S. Zhao, K. Tepe, I. Seskar, and D. Raychaudhuri. "Routing protocols for self-organizing hierarchical ad hoc wireless networks." In *IEEE Sarnoff Symposium*, 2003.

[7] L. Awuah, P. Burri and S. Puramshetty. "Comparing Cryptographic Protocols for Securing Wireless Ad-hoc Networks," Manuscript received at George Mason Univ. USA, 2005.

[8] P. Stanforth and V. Hasty. (2003, March 12) "Meshing Together: Advantages and Challenges of Deploying Ad-hoc Wireless Networks." *IEEE 802 LMSC Plenary Meeting.* [On-line]. Available: http://www.ieee802.org/meeting/archive/200303 [Dec 15, 2014].

[9] S. Dahlen (2003, Aug.). "How to deploy a VoIP gateway using a free software called Vocal on a Linux system." *Building VoIP using Vocal*. [On-line]. 1(1), Available: http://www.ece.fr:8000/~dahlen/Msc/Building_VoIP_using_Vocal_Issue1.1_(22-8-03)_final.pdf [Jan 12, 2014].

[10] J. Boleng. "Efficient Network Layer Addressing for Mobile Ad Hoc Networks," in *Proc. Int'l Conf. Wireless Networks*, 2002, pp. 271-277.

[11] H. Kim. "Dynamic Cooperative Communications in Wireless Ad-hoc Networks." Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, Jul. 22, 2008.

[12] W. PANICHPATTANAKUL. "Cooperative Communications in Ad Hoc Networks," Ph.D. dissertation, Poly technique National Institute Toulouse, Toulouse, France, Nov. 5, 2010.

[13] G. Cao and L. Yin. "Supporting Cooperative Caching in Ad-hoc networks." *IEEE Transaction on Mobile Computing*, vol. 5, pp. 77-89, Jan. 2006.

[14] E. Altman and T. Jimenez. Lecture Notes, Topic: "NS Simulator for Beginners." Sophia Antipolis, France, Dec. 4, 2003.

[15] E. Royer and C.K Toh. "A Review of Current Routing Protocols for Ad- Hoc Mobile Wireless Networks." *IEEE Personal Communications,* vol. 6(2), pp. 46-55, May 1999.

[16] D. Johnson and D. Maltz. "Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4." Internet: www.ietf.org/proceedings/76/RFCs/rfc4728.txt, Feb. 2007 [Jan. 5, 2014].

[17] C. Perkins, E. Royer and S. Das. "Ad hoc On-demand Distance Vector (AODV) Routing." Internet: www.ietf.org/rfc/rfc3561.txt, Feb. 2007 [Jan 5, 2014].

[18] C. Kozierok. "The TCP/IP Guide." Internet: www.tcpipguide.com, Sep. 20, 2005 [Jan. 11, 2014].

[19] G. Fairhurst. "Medium Access Control (MAC)." Internet: http://www.erg.abdn.ac.uk/users/gorry/course/lan-pages/mac.html, Mar. 10, 2009 [Jan. 11, 2014].

[20] M. Gong, B. Hart and S. Mao. "Advanced Wireless LAN Technologies: IEEE 802.11AC and Beyond." *Mobile Computing and Communications,* vol. 18(4), pp. 48-52, Oct. 2014.

[21] S. Banerji and R. Chowdhury. "On IEEE 802.11: Wireless LAN Technology." *Int. Journal of Mobile Network Communications & Telematics (IJMNCT),* vol. 3(4), Apr. 2013.

[22] B. Mitchell. "Introduction to MAC Addresses." Internet: http://compnetworking.about.com/od/networkprotocolsip/l/aa062202a.htm, Oct. 05, 2015 [Dec. 13, 2015].

[23] "Medium Access Control Protocol Data Units (MAC PDUs)." Internet: http://wimax-made-simple.blogspot.com/2009/10/medium-access-control-protocol-data.html, Oct. 8, 2009 [Mar. 20, 2015].

[24] W. Stallings. (2005). *Wireless communications and networks*. (2nd edition). [On-line]. Available: www.pearsonhighered.com/stallings/details5.html, [May 26, 2015].

[25] S. Banerji and R. Chowdhury. "On IEEE 802.11: Wireless LAN Technology." *Int. Journal of Mobile Network Communications & Telematics (IJMNCT),* vol. 3(4), Apr. 2013.

[26] J. Broachm, D. Maltz, D. Johnson, Y-C. Hu, J. Jetcheva, M. Thurston. "Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in *Proc. the 4th Annual ACM/IEEE Inter. Conf. on Mobile Computing and Networking (MobiCom TM 98),* 1988, pp. 85-97

[27] T. Holopainen. "IEEE 802.11 CSMA/CA Medium Access Protocol." http://www.netlab.tkk.fi/opetus/s38149/s02/reports/CSMA_CA_th.doc, 2002.

[28] D. Rao, K. Sreenu and P. Kalpana. "A Study on Dynamic Source Routing Protocol for Wireless Ad Hoc Networks." *The International Journal of Advanced Research in Computer and Communication Engineering*, vol. 1(8), pp. 522-29, Oct. 2012.

[29] D. Johnson, D. Maltz, J. Broch. *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks*, Adhoc Networking, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001, pp. 139-172.

[30] D. Maltz. "On-Demand Routing in Multi-Hop Wireless Mobile Ad-hoc Networks." Ph.D. dissertation, Carnegie Mellon University, USA, 2001.

[31] K. Fall and K. Varadhan. "The NS Manual." Internet: http://www.isi.edu/nsnam/ns/doc/, Nov. 5, 2011 [Aug. 17, 2013].

[32] N. Sarkar and R. McHaney. "Modelling and Simulation of IEEE 802.11 WLANs: a Case Study of a Network Simulator," in *Proc. of the 2006 Information Resources Management Association international conf.*, 2006, pp. 715-718.

[33] Y. Pan, "Design Routing Protocol Performance Comparison in NS2: AODV Comparing to DSR as Example," Dept. of CS, SUNY Binghamton, Vestal NY 13850, 2006.

[34] J. Chung and M. Claypool. "The Network Simulator NS-2." Internet: http://www.isi.edu/nsnam/ns, Nov. 5, 2011 [Dec. 10, 2013].

[35] S. Jun-Zhao, "Mobile Ad Hoc Networking: An Essential Technology for Pervasive Computing," *International Conf. on Info-tech and Info-net, Proc.*, 2001, pp. 316-321.

[36] L. Feeney and M. Nilsson. "Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment," in *Proc. of IEEE INFOCOM*, Anchorage, 2001.

[37] The Network Simulator NS-2. Internet: http://www.isi.edu/nsnam/ns, Nov. 5, 2011 [Dec. 10, 2013].

[38] A. Nosratinia, T. Hunter, A. Hedayat. "Cooperative Communication in Wireless Networks." *IEEE Communications Magazine*, vol. pp. 76-80, Oct. 2004.

[39] A. Scaglione, D. Goeckel and J. Laneman. "Cooperative Communications in Mobile Ad Hoc Networks." *IEEE Signal Processing Magazine*, vol. 23(5), pp. 18-29, Sep. 2006.

[40] W. Yu, R. Liu, "Secure Cooperative Mobile Ad Hoc Networks against Injecting Traffic Attacks." *Proc. IEEE Int'l Conf. Sensor and Ad Hoc Comm. and Networks (SECON)*, 2005.

[41] I. Hussain and K. Swapna. “Cooperative MIMO Communications in Wireless Ad-hoc Rayleigh Fading Networks.” *International Journal of Smart Sensors and Ad-hoc Networks (IJSSAN)*, vol. 1(4), pp. 15-21, 2012.

[42] Q. Guan, F. Yu, S. Jiang, V. Leung and H. Mehrvar. “Topology Control in Mobile Ad Hoc Networks with Cooperative Communications.” *IEEE Wireless Communication*, vol. 19, pp. 74-79, 2012.

[43] W. Su, A. Sadek, K. Liu. "Cooperative Communication Protocols in Wireless Networks: Performance Analysis and Optimum Power Allocation." *Wireless Personal Communication*, vol. 44, pp. 181-217, Jan. 2008.

[44] C. Perkins, E. Royer and S. Das. “Ad hoc On-demand Distance Vector (AODV) Routing.” Internet: www.ietf.org/rfc/rfc3561.txt, Feb. 2007 [Jan 5, 2014].

[45] C. A. Balanis, "Antenna Theory", John Wiley and Sons Inc., 2003.

[46] X. He and F. Li. “Cooperative RTS/CTS MAC with relay selection in distributed wireless networks,” in *Int. Conf. UMTW*, 2009, pp. 1-8.

[47] B. Mitchell. “Wireless Standards 802.11a, 802.11b/g/n, and 802.11ac” Internet: https://www.lifewire.com/wireless-standards-802-11a-802-11b-g-n-and-802-11ac-816553. Feb 23, 2016 [Mar. 08, 2017].