# Co-operative Communication:Cross-layer Approach

## Major Project Report

Submitted in partial fulfillment of the requirements

For the degree of

**Master of Technology**

**In**

**Electronics & Communication Engineering**

**(Communication Engineering)**

By


**Fichadiya Vishal Vinodbhai**

**(09MECC06)**



**Department of Electronics & Communication Engineering**

**Institute of Technology**

**Nirma University**

**Ahmedabad-382 481**


**May 2011**

# Co-operative Communication:Cross-layer Approach

**Major Project Report**

Submitted in partial fulfillment of the requirements

For the degree of

**Master of Technology**

**In**

**Electronics & Communication Engineering**

**(Communication Engineering)**

By

**Fichadiya Vishal Vinodbhai**

**(09MECC06)**

Under the Guidance of

**Dr. D. K. Kothari**

**Department of Electronics & Communication Engineering**

**Institute of Technology**

**Nirma University**

**Ahmedabad-382 481**

**May 2011**

# Declaration

This is to certify that

i) The thesis comprises my original work towards the degree of Master of Technology in Communication Engineering at Nirma University and has not been submitted elsewhere for a degree.

ii) Due acknowledgement has been made in the text to all other material used.

**Fichadiya Vishal Vinodbhai**

# Certificate

This is to certify that the Major Project entitled **"Co-operative Communication: Cross-layer Approach"** submitted by **Fichadiya Vishal Vinodbhai(09MECC06)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Communication Engineering of Nirma University, Ahmedabad is the record of work carried out by him under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of our knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Date:

Place: Ahmedabad

| **Guide & Program Co-ordinator** | **HOD** |
|---|---|
| Dr. D. K. Kothari<br>(Professor, EC) | Prof. A. S. Ranade<br>(Professor, EC) |

**Director**

Dr K Kotecha

(Director, IT-NU)

# Abstract

Exploding demand for a growing number of wireless applications has fueled significant development of wireless networks, especially several generations of cellular voice and data networks and, more recently, ad-hoc data networks for wireless computer, home, and personal networking. To overcome the problems wireless medium (fading,interference,etc.), MIMO system can be used but due to size, cost, or hardware limitations, a wireless node may not be able to support multiple transmit antennas. So, the cooperative communication can be used as an alternative to MIMO. Due to the broadcast nature of wireless signals, a wireless transmission intended for a particular destination station can be overheard by other neighboring stations. A focus in cooperative communications is to achieve spatial diversity gains by requiring these neighboring stations to retransmit the overheard information to the final destination. Layered protocol design provides reasonable performance in wired networks, where individual layer dynamics are limited and the system bandwidth and power are relatively unconstrained. However, it may not be efficient for wireless networks, since the large random dynamics in the wireless link strongly affect performance and design of Network layer protocols. In this dissertation, We have discussed CoopMAC, Cross-layer Protocol used for Cooperative Communication. In CoopMAC Protocol, high data rate stations assist low data rate stations in their transmission by forwarding their traffic. The CoopMAC protocol is simple and backward compatible with the legacy IEEE 802.11 system. An event driven simulator is developed and performance evaluation is carried out in terms of throoughput and delay. The performance is evaluated considering different network scenario for IEEE 802.11 and cooopMAC protocol and a comparison between them is presented. The CoopMAC protocol performs better than its legacy IEEE 802.11 in terms of throughput and delay and achieves higher spatial diversity.

# Acknowledgements

I would like to express my gratitude and sincere thanks to Prof. A. S. Ranade Head of Electrical Engineering Department. I am deeply indebted to Dr. D. K. Kothari,my thesis supervisor and Coordinator M.Tech Communication Engineering program for allowing me to undertake this thesis work and for his guidelines during the review process. He has devoted significant amount of his valuable time to plan and discuss the thesis work.

I am also thankful to Prof. M. A.Upadhyay, Prof.Y. N. Trivedi, Prof. M. R. Naik, Prof. S. H. Gajjar, and all the other faculty members who criticized me at each review and suggested me to do further changes.

I wish to thank my friends of my class for their delightful company which kept me in good humor throughout the year.

Last, but not the least, no words are enough to acknowledge constant support and sacrifices of my family members because of whom I am able to complete the degree program successfully.

**- Fichadiya Vishal Vinodbhai**
**09MECC06**

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Due to significant development in wireless network, espacially in cellular voice and data networks, more in ad-hoc networks, the demand for a wireless application is increased.In this chapter, we first introduce some background on cooperation and cross-layer design in wireless networks. Then, we describe the motivation and contributions of this dissertation. Finally, the outline of this dissertation is provided.

## 1.1 Cooperative Commmunication

In last decade, there is a tremendous growth in wireless application[1]. However, the wireless medium suffers from many limitation such as multipath fading, attenuation, signal loss, cost, size, complexity, limited resources (power, bandwidth), battery life etc. To overcome the problem of fading, attenuation, signal loss, there are many diversity techniques such as time diversity, frequency diversity, space diversity, transmit diversity, receiver diversity, MIMO etc. are used. The advantages of MIMO systems have been widely acknowledged, to the extent that certain transmit diversity methods (i.e., Alamouti signaling) have been incorporated into wireless standards. Although transmit diversity is clearly advantageous on a cellular base station, it may not be practical for other scenarios. Specifically, due to size, cost, or hardware limitations, a wireless agent may not be able to support multiple transmit antennas. Examples include most handsets (size) or the nodes in a wireless sensor network (size, power). So, the alternative for this MIMO system is the Cooperative communication[2].

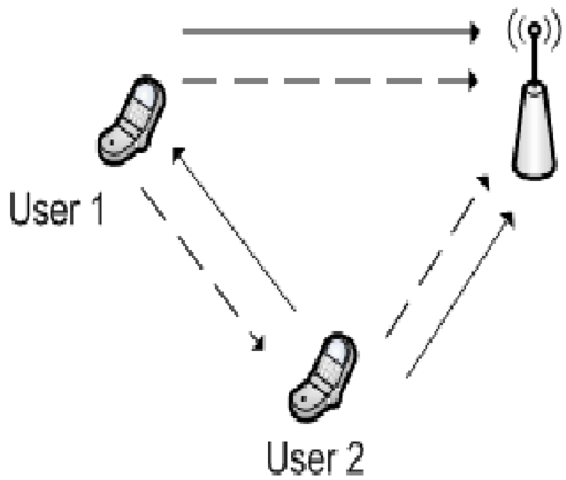



Figure 1.1: Cooperative communication

Cooperative communication allow single-antenna mobiles to reap some of the benefits of MIMO systems. The basic idea is that single-antenna mobiles in a multi-user scenario can "share" their antennas in a manner that creates a virtual MIMO system[2]. The mobile wireless channel suffers from fading, meaning that the signal attenuation can vary significantly over the course of a given transmission. Transmitting independent copies of the signal generates diversity and can effectively combat the deleterious effects of fading. In particular, spatial diversity is generated by transmitting signals from different locations, thus allowing independently faded versions of the signal at the receiver. Cooperative communication generates this diversity in a new and interesting way.

The figure 1.1 shows two mobile agents communicating with the same destination. Each mobile has one antenna and cannot individually generate spatial diversity.

However, it may be possible for one mobile to receive the other, in which case it can forward some version of ”overheard” information along with its own data. Because the fading paths from two mobiles are statistically independent, this generates spatial diversity[3].

In the figure 1.1, we use icons resembling base stations or handsets, but this is only a convenient graphical representation. The idea of cooperation is general, and perhaps even more suitable to ad hoc wireless networks and wireless sensor networks than cellular networks.[2]

In cooperative wireless communication, we are concerned with a wireless network, of the cellular or ad hoc variety, where the wireless agents, which we call users, may increase their effective quality of service (measured at the physical layer by bit error rates, block error rates, or outage probability) via cooperation. In a cooperative communication system, each wireless user is assumed to transmit data as well as act as a cooperative agent for another user.[3]

## 1.2 Cross layer Design for Wireless Network

Traditionally, protocol design in wired and wireless networks was primarily based on layered approaches(OSI model) that facilitated standardization and implementation[8]. For example, the physical (PHY) layer is responsible for the reliable and efficient delivery of information bits, while the medium access control (MAC) layer is responsible for resource management among multiple users in the network. In layered protocols, each isolated layer in the protocol stack is designed and operated independently, with predefined interfaces between layers that are static and independent of network constraints and applications. Layered protocol design provides reasonable performance in wired networks, where individual layer dynamics are limited and the system band-

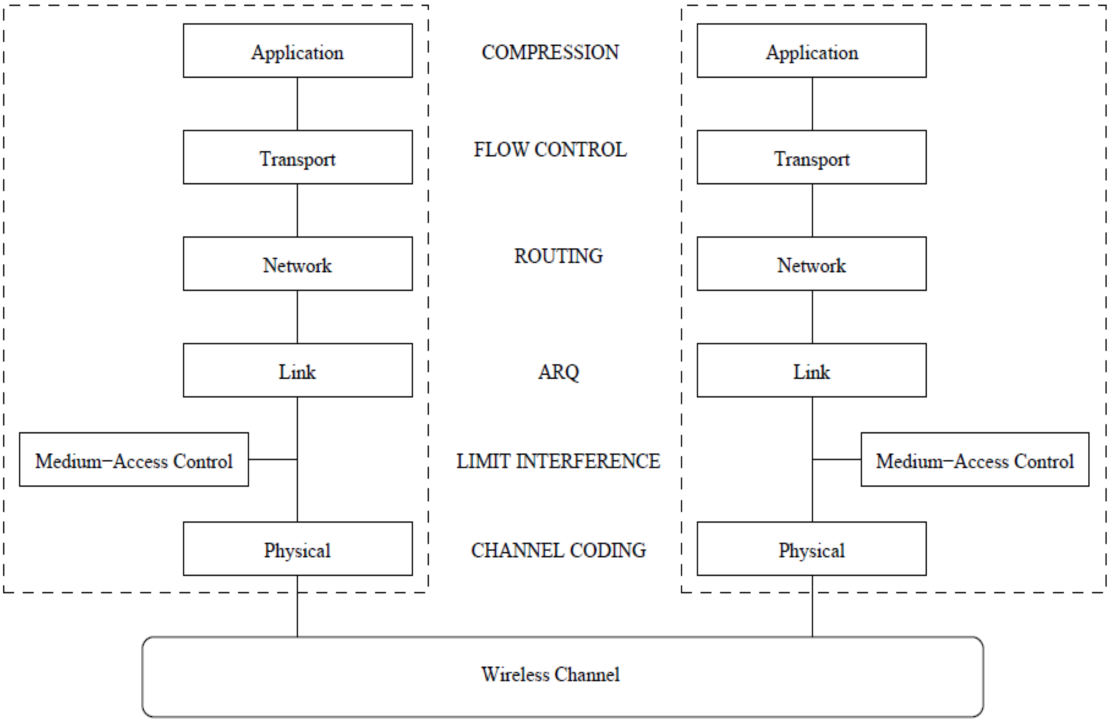



Figure 1.2: Layered protocol architecture

width and power are relatively unconstrained. However, it may not be efficient for wireless networks, since the large random dynamics in the wireless link (e.g., fading, Doppler-effect, and interference) strongly affect performance and design of network layer protocols Therefore, good and effective protocol designs are critical and imperative for wireless networks[8]. The layered Protocol architecture is as shown in 1.2.

Recently, cross-layer design has emerged as a preferable approach to enhance the performance of wireless networks. In the survey paper[8][10]. cross-layer design is defined as **"protocol design by the violation of a reference layered communication architecture"**. Examples of violation of a layered architecture include creating new interfaces between layers, redefining the layer boundaries, designing protocol at a layer based on the details of how another layer is designed, joint tuning of parameters across layers, and so on". In relevant research work, cross-layer design is often referred to as exploiting the interaction and performing joint optimization and design across multiple layers under given resource constraints. Examples of cross-layer design include joint design for multiuser scheduling, joint design for throughput maximization, and joint routing-MAC-link optimization.

Cooperation and cross-layer design are two emerging techniques for wireless networks.

## 1.3 Contribution of this Dissertation

Signal loss in wireless networks due to distance and fading severely impairs system performance. In most implementations, transmissions received by stations other than the intended receiver will be discarded. To exploit the broadcast nature of wireless channels, work on cooperative communication has shown that additional "cooperative" nodes, which overhear the transmission from the sender and then participate in additional transmission, can provide space diversity for the system and hence, the system performance is improved. The layered protocol architecute is not well suitable for wireless networks, we have studied a cross-layer protocol (MAC layer) called CoopMAC, which is based on IEEE 802.11 standards. The IEEE 802.11 standards has been implemented in NS-2. We have modified mac layer functionality of IEEE 802.11 to accomodate the CoopMAC protocol. Simulation are carried out in NS-2 to study the effect of this on system performance. The Simulations are carried out considering different network scenarios such as fixed node position and node movements. Finally performance comparison is done between IEEE 802.11 and the CoopMAC protocol.

## 1.4 Thesis Organization

The rest of the thesis is organized as follows.

Chapter 2 describes Cooperative communication, its working and different methods to achieve the cooperative communication. This chapter gives an insight into the concept behind the cooperative communication and its salient features.

Chapter 3 demonstrates cross layer approach and why it is required. It also gives details of one such Protocol CoopMAC used in the cooperative communication.

Chapter 4 gives the results of various simulations done for IEEE 802.11 with diffferent network scenarios. CoopMAC protocol with different configuration and comparison between them.

chpater 5 shows the results of various simulation done for CoopMAC protocol with different configuration and comparison between the CoopMAC protocol and its legacy IEEE 802.11.

Finally, in chapter 6 concluding remarks and scope for future work is presented.

# Chapter 2

# Overview of Cooperative Communication

## 2.1 Background

In wireless telecommunications, multipath is the propagation phenomenon that results in radio signals reaching the receiving antenna by two or more paths. Causes of multipath include atmospheric ducting, ionospheric reflection and refraction, and reflection from water bodies and terrestrial objects such as mountains and buildings. The effects of multipath include constructive and destructive interference, and phase shifting of the signal.

The presence of reflectors in the environment surrounding a transmitter and receiver create multiple paths that a transmitted signal can traverse. As a result, the receiver sees the superposition of multiple copies of the transmitted signal, each traversing a different path. Each signal copy will experience differences in attenuation, delay and phase shift while travelling from the source to the receiver. This can result in either constructive or destructive interference, amplifying or attenuating the signal power seen at the receiver. Strong destructive interference is frequently referred to as a deep fade and may result in temporary failure of communication due to a severe drop in the channel signal-to-noise ratio. This effect is known as fading, which is shown in fig 2.1).

Fading can cause poor performance in a communication system because it can result in a loss of signal power without reducing the power of the noise. This signal loss can be over some or all of the signal bandwidth. Fading can also be a problem as it changes over time: communication systems are often designed to adapt to such impairments, but the fading can change faster than the adaptations can be made. In such cases, the probability of experiencing a fade (and associated bit errors as the signal-to-noise ratio drops) on the channel becomes the limiting factor in the performance of the link.[2]

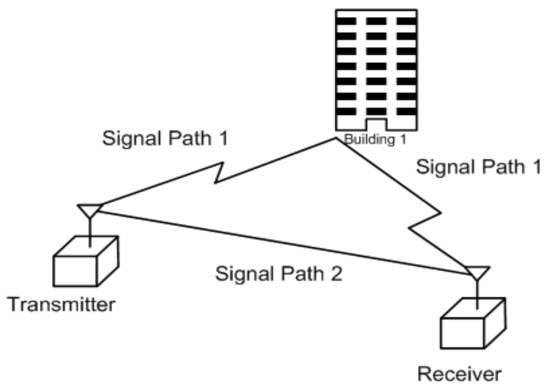



Figure 2.1: Multipath fading Phonomenon

The effects of fading can be combated by using diversity to transmit the signal over multiple channels that experience independent fading and coherently combining them at the receiver. The probability of experiencing a fade in this composite channel is then proportional to the probability that all the component channels simultaneously experience a fade, a much more unlikely event.

Diversity can be achieved in time, frequency, or space. Diversity techniques in wireless communications, whether based in time, frequency, or space, have long been used as a means to introduce reliability into an unreliable medium. Wireless links suffer from many impairments, including such effects as path loss, shadowing, and multipath fading.[5]

The various diversity schemes are classified as below:

**Time diversity** Multiple versions of the same signal are transmitted at different time instants. Alternatively, a redundant forward error correction code is added and the message is spread in time by means of bit-interleaving before it is transmitted. Thus, error bursts are avoided, which simplifies the error correction.

**Frequency diversity** The signal is transmitted using several frequency channels or spread over a wide spectrum that is affected by frequency-selective fading.

**Space diversity** The signal is transmitted over several different propagation paths. In the case of wired transmission, this can be achieved by transmitting via multiple wires. In the case of wireless transmission, it can be achieved by antenna diversity using multiple transmitter antennas (transmit diversity) and/or multiple receiving antennas (reception diversity). In the latter case, a diversity combining technique is applied before further signal processing takes place. If the antennas are far apart, for example at different cellular base station sites or WLAN access points, this is called macro diversity or site diversity. If the antennas are at a distance in the order of one wavelength, this is called micro diversity. A special case is phased antenna arrays, which also can be used for beam forming, MIMO channels and Space-time coding (STC).

**Polarization diversity** Multiple versions of a signal are transmitted and received via antennas with different polarization. A diversity combining technique is applied on the receiver side.

**Multiuser diversity** Multiuser diversity is obtained by opportunistic user scheduling at either the transmitter or the receiver. Opportunistic user scheduling is as follows: the transmit selects the best user among candidate receivers according to the qualities of each channel between the transmitter and each receiver. In FDD systems, a receiver must feedback the channel quality information to the transmitter with the limited level of resolution.

**Cooperative diversity** Achieves antenna diversity gain by using the cooperation of distributed antennas belonging to each node.

## 2.2 Co-operative Communication

Transmit diversity generally requires more than one antenna at the transmitter. However, many wireless devices are limited by size or hardware complexity to one antenna. Recently, a new class of methods called cooperative communication has been in use that enables single antenna mobiles in a multi-user environment to share their antennas and generate a virtual multiple- antenna transmitter that allows them to achieve transmit diversity.[2]

The advantages of multiple-input multiple-output (MIMO) systems, which is shown in figure 2.2, have been widely acknowledged; to the extent that certain transmit diversity methods have been incorporated into wireless standards. Although transmit diversity is clearly advantageous on a cellular base station, it may not be practical for other scenarios. Specifically, due to size, cost, or hardware limitations, a wireless agent may not be able to support multiple transmit antennas. Examples include most handsets (size) or the nodes in a wireless sensor network (size, power)[4].

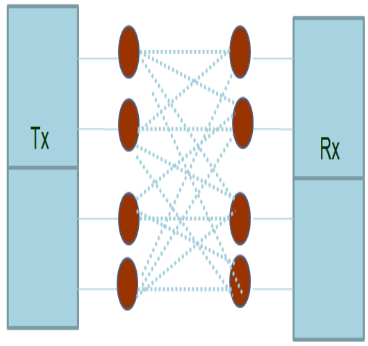



Figure 2.2: 4X4 MIMO

Transmitting independent copies of the signal generates diversity and can effectively combat the deleterious effects of fading. In particular, spatial diversity is generated by transmitting signals from different locations, thus allowing independently faded versions of the signal at the receiver. Cooperative communication generates this diversity in a new and interesting way.

**Cooperative communication**, which allow single-antenna mobiles to reap some of the benefits of MIMO systems. The basic idea is that single-antenna mobiles in a multi-user scenario can "share" their antennas in a manner that creates a virtual MIMO system.Fig 2.3 shows this concept.

The figure 2.3 shows two mobile agents communicating with the same destination. Each mobile has one antenna and cannot individually generate spatial diversity. However, it may be possible for one mobile to receive the other, in which case it can forward some version of "overheard" information along with its own data. Because the fading paths from two mobiles are statistically independent, this generates spatial diversity.

In this system, we are concerned with a wireless network, of the cellular or ad hoc

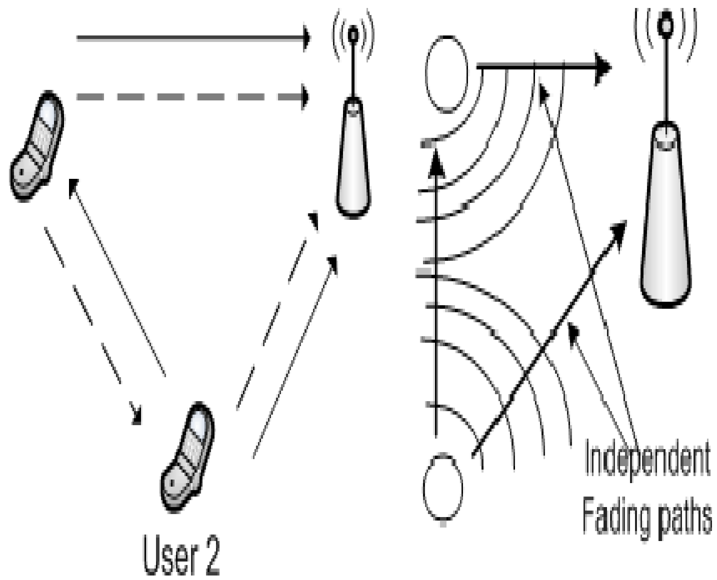



Figure 2.3: Cooperative Communication

variety, where the wireless agents, which we call users, may increase their effective quality of service via cooperation.

In cooperative communications, nodes utilize their bandwidth and energy resources to transmit not only their own information, but also the observed transmissions of their neighbors. In so doing, information for a given destination travels over multiple spatial paths in the network, experiencing independent fading realizations. These independent realizations allow severe impairments to be averaged out providing the communication link with protection. Thus Cooperative communication means working or acting together willingly for a common purpose or benefit.[4]

In general, cooperative relaying systems have a source node multicasting a message to a number of cooperative relays, which in turn resend a processed version to the intended destination node. The destination node combines the signal received from the relays, possibly also taking into account the source's original signal. Fig 2.4 shows

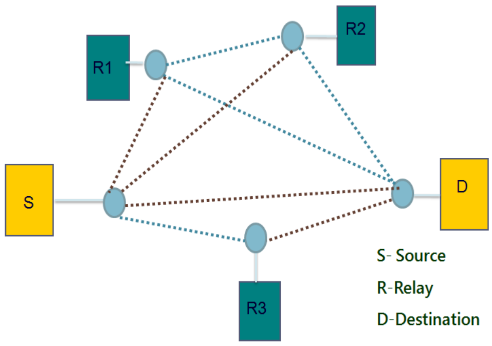



Figure 2.4: Scenario for cooperative communication

this concept. Cooperative diversity exploits two fundamentals features of wireless medium.

- its broadcast nature
- ability to achieve diversity through independent channels

There are three advantages from this:

**Diversity** This occurs because different paths are likely to fade independently. The impact of this is expected to be seen in the physical layer, in the design of a receiver that can exploit this diversity.

**Beamforming gain** The use of directed beams should improve the capacity on the individual wireless links. The gains may be particularly significant if space-time coding schemes are used.

**Interference mitigation** A protocol that takes advantage of the wireless channel and the antennas and receivers available could achieve a substantial gain in system throughput by optimizing the processing done in the cooperative relays and in the scheduling of retransmissions by the relays so as to minimize mutual interference and facilitate information transmission by cooperation.

The cooperative communication widely used in wireless ad-hoc network such as wireless sensor network and in cellular network. By using cooperative communication, multiple virtual-antenna transmitter can be considered, e.g. in a cellular networks. A relay channel is a three terminal network consisting of a source, a relay, and a destination. However, this concept can be widely extended to larger network configurations In wireless ad-hoc network, collection of wireless nodes are distributed in space. These nodes cooperate by retransmitting each others signals to other nodes. Thus the collection of nodes becomes a "distributed antenna array". The relay nodes simply amplify and forwarded the signal to the receiver. In this system, the wireless agents (users) may increase their effective quality of service via cooperation[7].

## 2.3 Relay Channel Concept

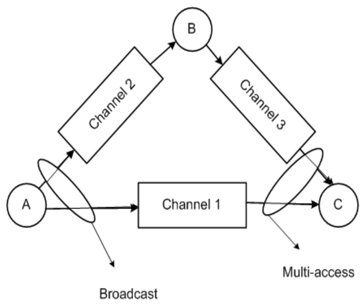



Figure 2.5: Relay channel

The basic ideas behind cooperative communication can be traced back to the groundbreaking work of Cover and El Gamal[9] on the information theoretic properties of the relay channel. This work analyzed the capacity of the three-node network consisting of a source, a destination, and a relay. It was assumed that all nodes operate in the same band, so the system can be decomposed into a broadcast channel from the viewpoint of the source and a multiple access channel from the viewpoint of the destination.

A relay channel is a three-terminal network consisting of a source, a relay and a destination[9]. The source broadcast to both relay and destination. Also, the relay forwards the received message to the destination. Relay systems can achieve distributed spatial diversity in wireless networks of single-antenna devise transmitting over quasi-static fading channel. Relaying can be used to form virtual antenna array. The strategy of cooperative diversity can be exploited by exchanging the role of source and relay.

## 2.4 Various Signaling Methods used for Cooperative Communication

There are mainly three type of signaling methods are used in cooperative communication, which are as listed below.[7].

- DETECT AND FORWARD METHOD
- AMPLIFY-AND-FORWARD METHOD
- CODED COOPERATION
- ESTIMATE AND FORWARD
- COMPRESS AND FORWARD

### 2.4.1 Detect and Forward Method

Figure 2.7 shows this method. This method is perhaps closest to the idea of a traditional relay. In this method a user attempts to detect the partner's bits and then retransmits the detected bits. The partners may be assigned mutually by the base

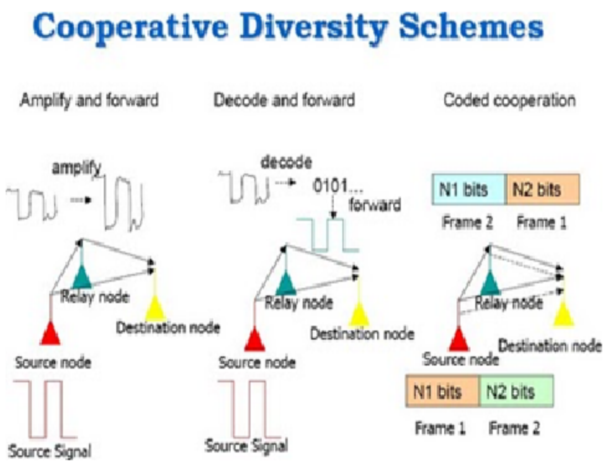



Figure 2.6: various cooperative diversity signaling methods

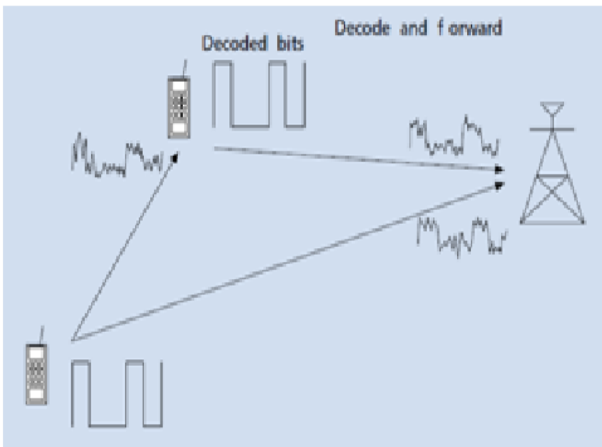



Figure 2.7: Detect and forward method[2]

station, or via some other technique. As shown in figure 2.7, two users partnering with each other are considered, but in reality the only important factor is that each user has a partner that provides a second (diversity) data path. The easiest way to visualize this is via pairs, but it is also possible to achieve the same effect via other partnership topologies that remove the strict constraint of pairing. Partner assignment is an important issues in this case.
This signaling has the advantage of simplicity and adaptability to channel conditions. This method has following disadvantages.

- First, it is possible that detection by the partner is unsuccessful, in which case cooperation is of no use at the base station.
- Also, the base station needs to know the error characteristics of the inter user channel for optimal decoding.

The problem of error propagation can be overcomed by hybrid decode and- forward method. In this method, at times when the fading channel has high instantaneous signal-tonoise ratio (SNR), users detect and forward their partners' data, but when the channel has low SNR, users revert to a non cooperative mode.

### 2.4.2 Amplify and Forward Method

Amplify and Forward method, as shown in figure 2.8, is the simple cooperative signalling method. Each user in this method receives a noisy version of the signal transmitted by its partner. The user then amplifies and retransmits this noisy version. The base station combines the information sent by the user and partner, and makes a final decision on the transmitted bit. Although noise is amplified by cooperation, the base station receives two independently faded versions of the signal and can make better decisions on the detection of information. It has been shown in literature that

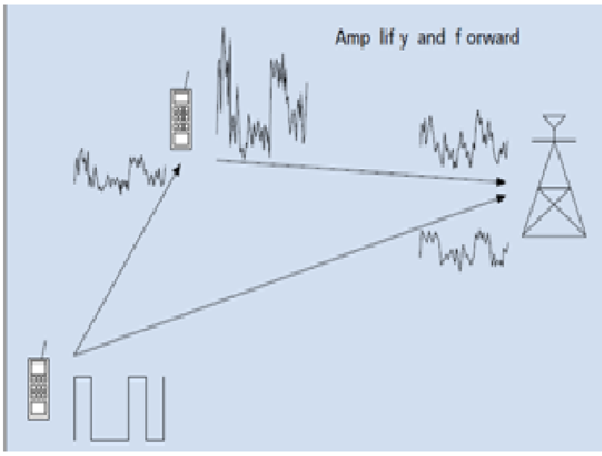



Figure 2.8: amplify and forward method[2]

for the two-user case, this method achieves diversity order of two, which is the best possible outcome at high SNR[5]. This method has following main advantages.

- Achieve full diversity
- Performance better than direct transmission and decode-and-forward
- Achieve the capacity when number of relays tend to infinity

### 2.4.3 Coded Cooperation

Coded cooperation, as shown in figure 2.9, is a method that integrates cooperation into channel coding. Coded cooperation works by sending different portions of each user's code word via two independent fading paths. The basic idea is that each user tries to transmit incremental redundancy to its partner. Whenever that is not possible, the users automatically revert to a non cooperative mode. The key to the efficiency of coded cooperation is that all this is managed automatically through code design, with no feedback between the users[5].

The users divide their source data into blocks that are augmented with cyclic

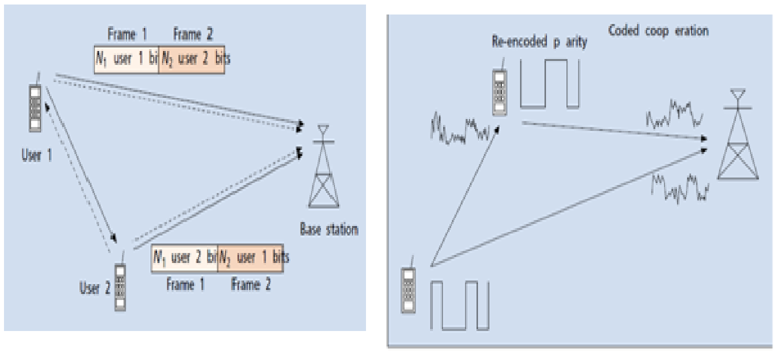



Figure 2.9: Coded cooperation method[2]

redundancy check (CRC) code. In coded cooperation, each of the users' data is encoded into a codeword that is partitioned into two segments, containing N1 bits and N2 bits, respectively. For example, consider that the original codeword has N1 + N2 bits; puncturing this codeword down to N1 bits, we obtain the first partition, which itself is a valid (weaker) codeword. The remaining N2 bits in this example are the puncture bits.

Likewise, the data transmission period for each user is divided into two time segments of N1 and N2 bit intervals, respectively. We call these time intervals frames. For the first frame, each user transmits a code word consisting of the N1-bit code partition. Each user also attempts to decode the transmission of its partner. If this attempt is successful (determined by checking the CRC code), in the second frame the user calculates and transmits the second code partition of its partner, containing N2 code bits. Otherwise, the user transmits its own second partition, again containing N2 bits. Thus, each user always transmits a total of N = N1 + N2 bits per source block over the two frames. Figure 2.9 illustrates the coded cooperation framework.

In general, various channel coding methods can be used within this coded cooperation framework. For example, the overall code may be a block or convolution code, or a combination of both. The code bits for the two frames may be selected through puncturing, product codes, or other forms of concatenation. The users act independently in the second frame, with no knowledge of whether their own first frame was correctly decoded. As a result, there are four possible cooperative cases for the transmission of the second frame: both users cooperate, neither user cooperates, user 1 cooperates and user 2 does not, and vice versa.

## 2.5 Four Possible Cooperative modes

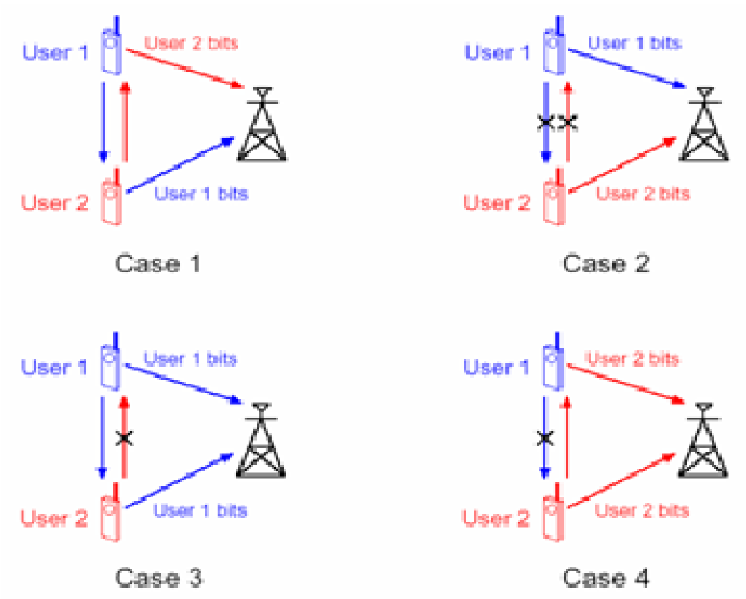



Figure 2.10: Four Possible Cooperative modes

In cooperative communication, whether the user cooperates each other fully, partially or not cooperating -based on this four modes are possible as shown in figure 2.10.

- Both users are cooperating each other.
- Both users are not cooperating each other.
- One user is cooperating, but other user is not cooperating.

## 2.6 Summary

In this chapter, various aspects of cooperative communication is described. Cooperative coommunication uses the broadcast nature of nodes for forwarding data of source to the destination. It achieves higher spatial diversity and increase the system performances. It reduces the complexity required for each node as in case of MIMO, by creating Virtual MIMO. Various signalling methods used for cooperative communication has been described.

# Chapter 3

# Cross layer Design approach

Various OSI layers and network functions must be considered together while designing efficient networks to support new multimedia services. Use of cross layer approach challenging the traditional OSI layered design, which is in use. Ad hoc network has emerged as an important trend of future wireless system that will provide ubiquitous wireless access. There is a tremendous growth in wireless application, espacially in celluar voice and data networks, more recently ad-hoc networks for wireless computer, home and personal networking. In this chapter, one such Cross-layer Protocol called, CoopMAC is presented.

## 3.1 The CoopMAC Protocol

Wireless networks that provide multi-rate support give the stations the ability to adapt their transmission rate to the the link quality in order to make their transmissions more reliable. Thus, stations that experience poor channel conditions tend to use lower transmission rates and vice versa [10]. There are two cases where stations will decide to transmit at a low transmission rate:

- When a station experiences a bad channel due to its distance from the Access Point.

- In an indoor environment with strong shadowing and fading effects, a station may experience a low quality channel at some spots in the coverage area.

In paper [10], it has been demonstrated that the presence of a few low data rate stations will have an adverse effect on the overall throughput of the network. For Example, the presence of stations at 1 Mbps reduces the average throughput of all the stations in the network with that of stations at 11Mbps. This is because a 1 Mbps station takes roughly 11 times more transmission time than an 11 Mbps station to transmit the same number of bits. One of the aims of cooperation at MAC layer is that it allows the high rate stations to help stations that can only sustain a low data rate to mitigate this effect.

The CoopMAC protocol is based on a simple but efficient cooperative scheme where intermediate stations between the low speed station and the Access Point can act as helpers in the transmission process. Instead of having a slow station transmitting its frame directly to the Access Point, an alternative route through a high speed station is used sending the frame in a two-hop manner. Thus, instead of using one low data rate transmission, two high data rate transmissions are used, decreasing the amount of time the channel is occupied for that particular transmission. This cooperative protocol is compatible with the widely used IEEE 802.11 standard and hence can be deployed incrementally[11].

The cross-layer protocol, which is called CoopMAC, enables cooperation in 802.11 networks.There are two potential costs for the helper stations introduced by the CoopMAC protocol. An opportunity cost, which is due to the transmission opportunities a helper may have to lose in order to forward data for another station, and an actual cost, which is the energy expense involved in transmitting data for other stations. It will become obvious that there is no opportunity cost because the helper nodes are allowed to access the channel without contention when forwarding data for other stations. It is shown in literature that for helping nodes participating in CoopMAC protocol, there is substantial saving of energy. This is due to fact the amount of

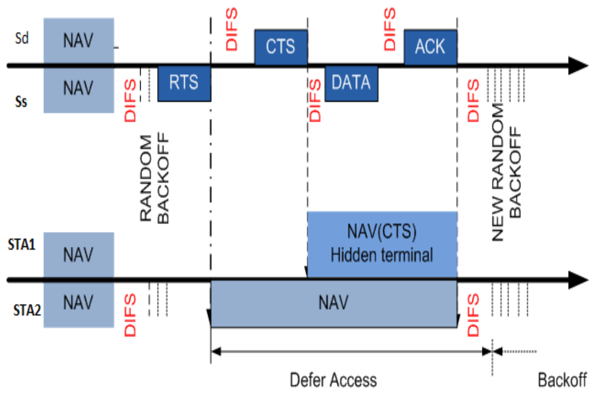



Figure 3.1: NAV mechanism defined in IEEE 802.11

cooperation which is idle time of helper node befor this get access to channel for its own transmission.[11]

The CoopMAC protocol maintains backward compatibility with the legacy IEEE 802.11 distributed coordination function (DCF) describe this protocol.As the coop-MAC protocol is based on IEEE 802.11, so first we describe it shortly and then we will describe this protocol.

### 3.1.1 The IEEE 802.11 Protocol

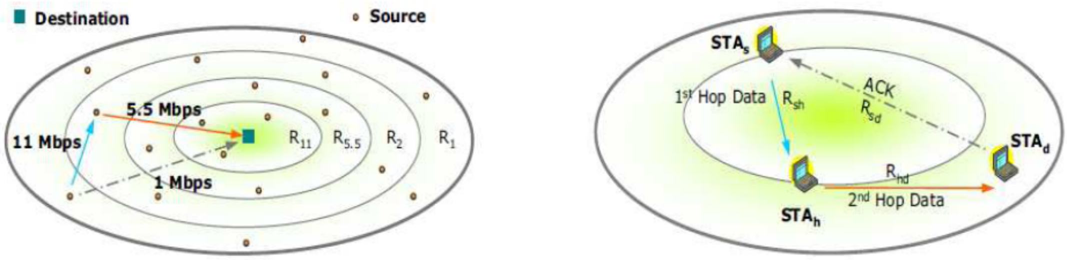
The IEEE 802.11 protocol employs carrier sense multiple access with collision avoidance (CSMA/CA) as its medium access protocol for the distributed coordination function (DCF) mode. In this mode, each station (STA) can initiate a data transmission by itself. Channel sensing before packet transmission is essential to avoid collisions. If one station has date packet to send, it will first sense the channel to make sure the channel is clear before the actual transmission starts. Since not all stations can hear each other, even if the channel is sensed to be free, a collision may occur. Thus virtual carrier sensing is also employed with the use of the Request To Send (RTS) and Clear To Send (CTS) frames to reserve channel time for the transmitting stations. These two control frames broadcast the channel reservation information to the whole network. Any station will be able to hear at least one of these control packets and use them to calculate the time needed for the data packet transmission. A Network Allocation Vector (NAV) is used by all the stations to discover the time for which the channel is going to be free.

The band that 802.11b uses is the 2.4GHz band. In the Physical layer, it deploys three different modulation schemes to support 4 different transmission rates, 1, 2, 5.5 and 11Mbps.

**Multirate Capability**

Before going to the protocol details of CoopMAC, the multi-rate capability of IEEE 802.11b requires a brief discussion, as it is required for understanding cooperation at the MAC layer. In order to deliver an acceptable frame error rate (FER), packets in IEEE 802.11 can be transmitted at different bit rates, which are adaptive to the channel quality. In general, the transmission rate is essentially determined by the path loss and instantaneous channel fading conditions[14]. For IEEE 802.11b, in particular, four different rates are supported over the corresponding ranges, as depicted in Figure 3.2.

Another key observation conveyed by Figure 3.2 is that a source station that is far away from the destination may persistently experience a poor wireless channel, resulting in a rate as low as 1Mbps for direct transmission over an extended period of time. If there exists some neighbor who in the meantime can sustain higher transmission rates (e.g., 11Mbps and 5.5Mbps in Figure 3.2) between itself and both the source and the intended destination, the source station can enlist the neighbor to cooperate and forward the traffic on its behalf to the destination, yielding a much higher equivalent rate. With the simple participation of a neighboring station in the cooperative forwarding, the aggregate network performance would witness a dramatic




(a) Transmission range versus rate. The 11 Mbps, 5.5 Mbps, 2 Mbps and 1 Mbps regions are denoted by $R_{11}$, $R_{5.5}$, $R_2$ and $R_1$, respectively.

(b) Data plane operation of CoopMAC.

Figure 3.2: Cooperation at MAC layer

improvement, which justifies and motivates the introduction of cooperation into the MAC layer[13]. Thus to enable cooperation in the IEEE 802.11 MAC layer, the RTS/CTS signaling defined in IEEE 802.11 can be extended to a 3-way handshake in CoopMAC to further facilitate the ensuing cooperative data exchange.

### 3.1.2 CoopMAC Protocol

The CoopMAC protocol, is a protocol based on cross layer approach. It also exploits the concept of Cooperative Communication. Figure 3.3 shows the message flow (Handshaking signalling and data transfer) and figure 3.4shos the NAV(Network Allocation Vector) setting with helper & without helper.

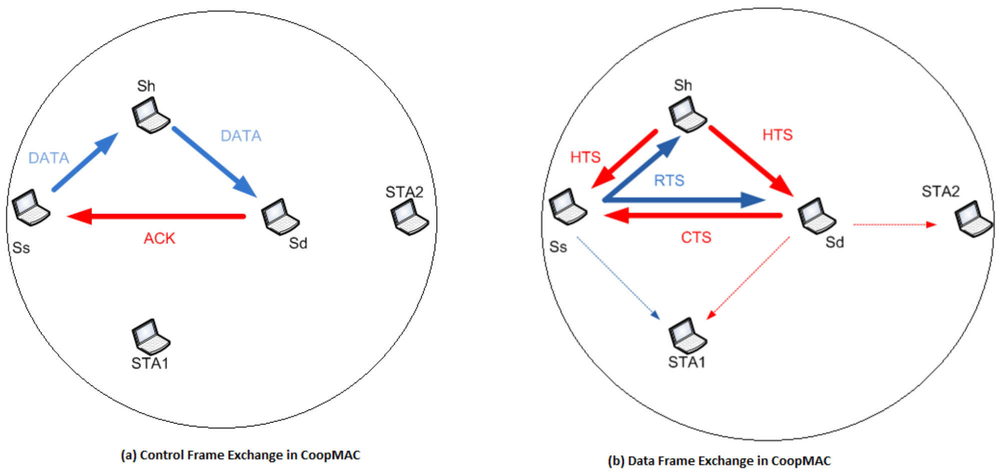



Figure 3.3: Message flow in CoopMAC

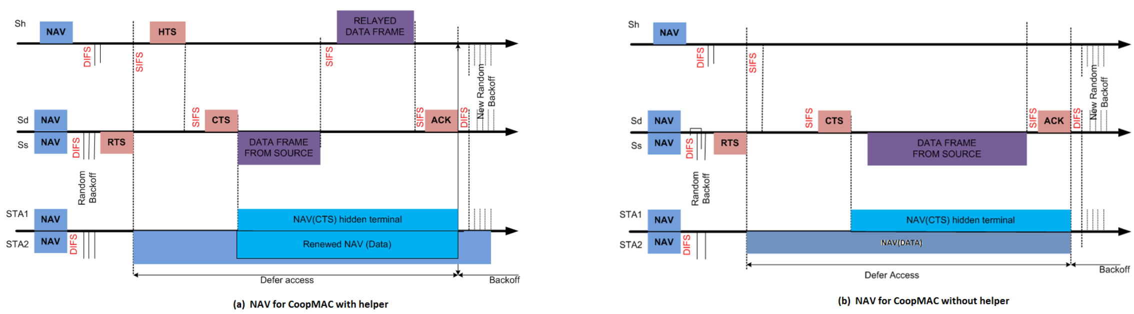



Figure 3.4: NAV settings in CoopMAC

### 3.1.3 The message flow in CoopMAC protocol

The message flow control in CoopMAC protocol is explained as under.

- When a source node has a new MAC protocol data unit (MPDU) to send, it can either transmit directly to the destination, or use an intermediate helper for relaying, whichever consumes less total air time. The air time is compared using cached information on the feasible data rates between the three nodes. The feasible data rate is the largest data rate that guarantees a predetermined average error rate threshold for an average channel SNR.

- Beyond its normal function, a request to send (RTS) message is also used by CoopMAC to notify the node that has been selected for cooperation. Moreover, CoopMAC introduces a new message called helper-ready to send (HTS), which is used by the helper to indicate its availability after it receives the RTS message from the source. If the destination hears the HTS message, it issues a clear to send (CTS) message to reserve channel time for a two-hop transmission. Otherwise, it still sends out the CTS, but only reserves channel time for a direct transmission.

- If both HTS and CTS are received at the source, the data packet should be transmitted to the relay first, and then forwarded to the destination by the relay.

- A normal ACK is used to acknowledge a correct reception, regardless of whether the packet is forwarded by the relay, or is directly transmitted from the source. If necessary, retransmission is attempted. It is crucial that each node obtains and constantly updates its information about the availability of potential relays. The CoopMAC protocol deals with this issue mainly through maintaining a table called the CoopTable[11], in its management plane. The CoopTable is as shown in Table 3.1. Each entry in the CoopTable corresponds to a potential relay, and contains such information as the ID (e.g., 48-bit MAC address) of the potential relay, the latest time at which a packet from that potential relay is overheard by the source, and the data rate used for direct transmission between the potential relay and destination, and between the current node and the potential relay. A set of protocols have been defined in CoopMAC to properly establish, manage, and update the table in a timely manner.

- Due to the broadcast nature of the channel, the destination will receive the signals transmitted by both the source and the relay. If the destination is capable of combining these two copies to decode the original information, then cooperative diversity can be fully leveraged.

### 3.1.4 The CoopTable

Table 3.1: The CoopTable

| **ID(48Bits)** | **Time(8Bits)** | $R_{hd}$**(8Bits)** | $R_{hd}$**(8Bits)** | **NumOfFails** |
|---|---|---|---|---|
| MAC address of helper 1 | Time the last packet heard from helper 1 | Transmission rate between helper 1 and the destination | Transmission rate between the source and helper 1 | Count of sequential transmission failure |
| ....... | ....... | ....... | ....... | ....... |
| MAC address of helper N | Time the last packet heard from helper N | Transmission rate between helper N and the destination | Transmission rate between the source and helper N | Count of sequential transmission failure |

Each entry in the CoopTable, which corresponds to one candidate helper $STA_h$, is indexed by its MAC address. The values of $R_{hd}$ and $R_{sh}$ associated with $STA_h$ are stored in the third and fourth field of the CoopTable, respectively. The time at which the most recent packet is overheard from $STA_h$, is held in the second field called Timestamp[14]. The last field, Number of Failures, reflects the reliability of each helper, by recording the number of consecutive unsuccessful transmissions that use $STA_h$ as a helper.

### 3.1.5 Frame format

This section describes the various frames formats which are used in CoopMAC protocol. There are mainly four frames are used in this CoopMAC protocol.[11].

- CoopRTS(Request To Send)
- HTS(Helper Ready To Send)
- CTS(Clear To Send)
- ACK(Acknowledgement)

**The CoopRTS Frame** The frame format of CoopRTS is as under.

Table 3.2: The CoopRTS frame

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Frame Control | Duration | RA | TA | Helper ID | Rsh | Rhd | FCS |

Where, **Frame control** - The frame control for coopRTS is same as that of mac header of 802.11, which is shown in figure 3.5.

**Duration** - Time up to which stations are busy or time to transfer the data between the stations.(2bytes).

**RA** - Receiving station Address (6 bytes).

**TA** - Transmitting station Address (6 bytes).

**Rsh** - Rate between source and helper station.

**Rhd** - Rate between helper and destination station.

**Mac header for 802.11** The figure 3.5 shows the Mac header and frame control for 802.11.

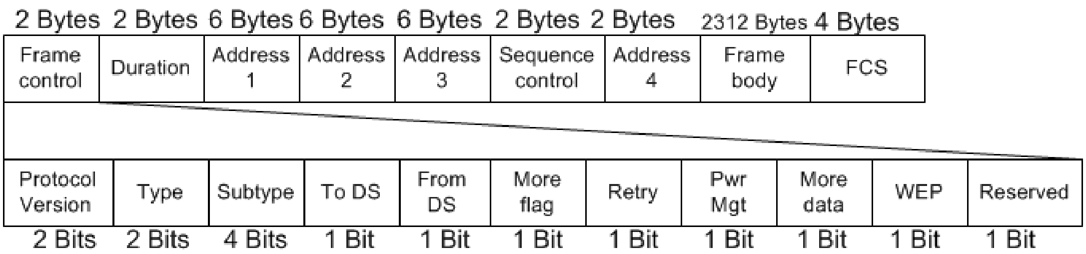



Figure 3.5: Frame Control for 802.11

Where, **Frame control** - This field contains following fields.

Table 3.3: The Framecontrol for MAC 802.11

| **Field** | **Explanation** |
|---|---|
| Version | Current version is 0 |
| Type | Type of information : management(00),control(01), or data(10) |
| Subtype | Subtype of each type |
| To DS | The frame is going to or coming to intercell distribution system |
| From DS | The frame is going to or coming to intercell distribution system |
| More flag | When set to 1, means more fragments |
| Retry | When set to 1, means retransmitted frame |
| Pwt Mgt | When set to 1, means station is in power management mode. |
| More data | When set to 1, means station has more data to send |
| WEP | Wired equivalent privacy (encryption implemented) |
| Rsvd | Reserved |

**Address1, Address2, Address3, Address 4** - These addresses are used for addressing mechanism.(each address of 6Bytes.) Different addressing mechanism is as shown in Table 3.6.

**Addressing mechanism** Table 3.4 and fig 3.6 shows the addressing mechanism for CoopMAC protocol.

Table 3.4: Addressing Mechanism for CoopMAC

| Scenario | Address1 | Address2 | Address4 |
|---|---|---|---|
| Source to Destination | Destination | Source | Not used |
| Source to Helper | Helper | Source | Destination |
| Helper to Destination | Destination | Source | Not used |

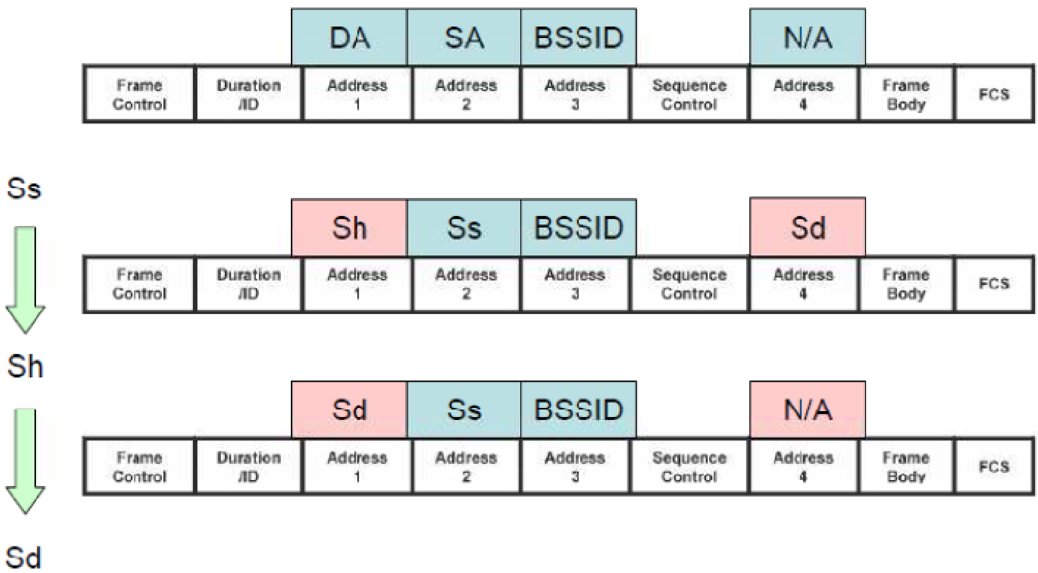



Figure 3.6: Addressing mechanism

To identify station that has been selected as a helper, the Address 4 field in the MAC header of data packet from STAs to STAh in CoopMAC should hold the MAC address of the final destination STAd, while the Address 1 field contains the MAC address of the selected helper STAh. When the packet is further forwarded by STAh to STAd, the helper will place the adderss of STAd in field Address 1, and leave the Address 4 unused.

### 3.1.6 Flowchart

The flowchart for CoopMAC protocol for Source(Ss),Helper(Sh),Destination(Sd) are shown in figures.[11]

### 3.5.2.1. Flow chart at Source station

- Whenever there is at least one packet buffered in the queue, $S_s$ should search for a helper candidate in the CoopTable. If a helper entry is successfully found, $S_s$ sends a CoopRTS message with the helper ID in the Address 1 field to specify the helper being selected. Besides, $R_{sh}$ and $R_{hd}$ also should be included in the corresponding fields of CoopRTS, indicating the expected data rates between $S_s$ and $S_h$, and between $S_h$ and $S_d$, respectively. If the table lookup yields a failure, the regular IEEE 802.11 MAC procedure for data transmission should be followed.

- If neither an HTS from $S_h$ nor a CTS from $S_d$ is heard after 2SIFS + CTS time, or a CTS is lost after an HTS was sent by the helper, $S_s$ should perform regular random backoff, as if it encountered a collision.

- If $S_s$ does not receive any HTS message from $S_h$, but does hear a CTS from $S_d$, it should then send the data, using a direction transmission. To update the CoopTable, $S_s$ should increment NumOfFailures, if the HTS message is not received after a SIFS time. If the value of NumOfFailures is greater than the threshold (i.e., 3), $S_s$ should remove the entry from CoopTable.

- If both HTS and CTS messages are received, $S_s$ sends the data to $S_h$ at the rate of $R_{sh}$, and set the ACK timeout.

- If an acknowledgment (ACK) is not received after an ACK timeout, $S_s$ should perform random backoff, following the legacy 802.11 protocol. Otherwise, $S_s$ should declare a success, and handle the next packet in its queue.

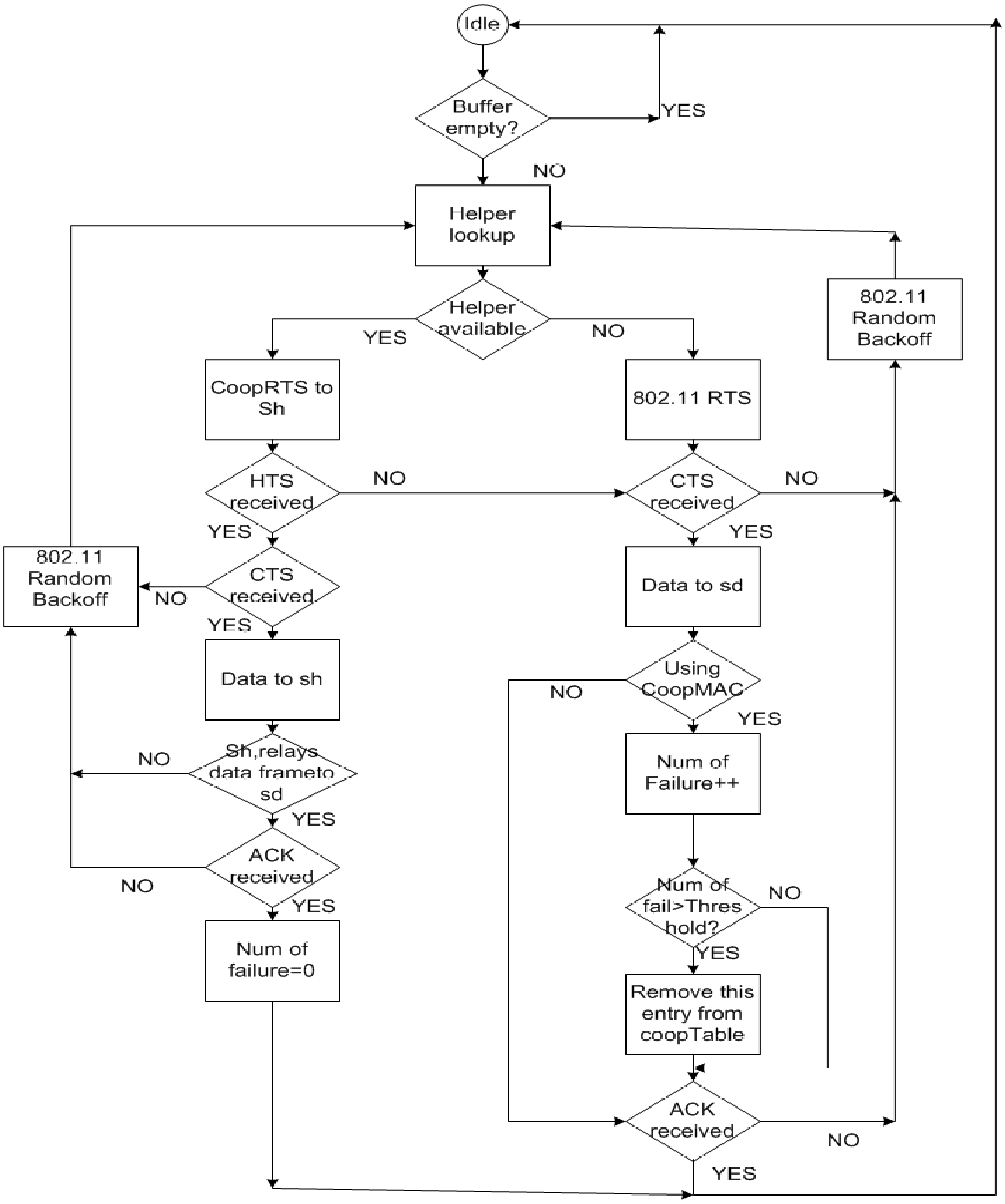



Figure 3.7: FLow Chart for Source Station

#### 3.5.2.2. Flow chart at Helper station

- If $S_h$ receives a CoopRTS message, whose Address 1 field contains its MAC address, then $S_h$ should verify whether the rate $R_{sh}$ between itself and $S_s$, and a rate Rhd between itself and $S_d$ suggested in the CoopRTS message are sustainable. If yes, it then sends an HTS message back to $S_s$, after a SIFS time, with the duration field.
- After sending the HTS to $S_s$, $S_h$ should run a timer of value TSIFS+TCTS, and expect a CTS from $S_d$. If $S_h$ does receive such a CTS, it then should wait for the data packet from $S_s$ to arrive SIFS time after the CTS message. If $S_h$ does not receive either the CTS message or the data packet as expected, it should assume that the data transmission was aborted, and revert to the initial state. When the data packet to be forwarded arrives, $S_h$ should forward the data packet to $S_d$ at the rate Rhd, a SIFS interval after the completion of the reception.
- If $S_h$ cannot support rates $R_{sh}$ and Rhd, $S_h$ simply goes back to the initial state.

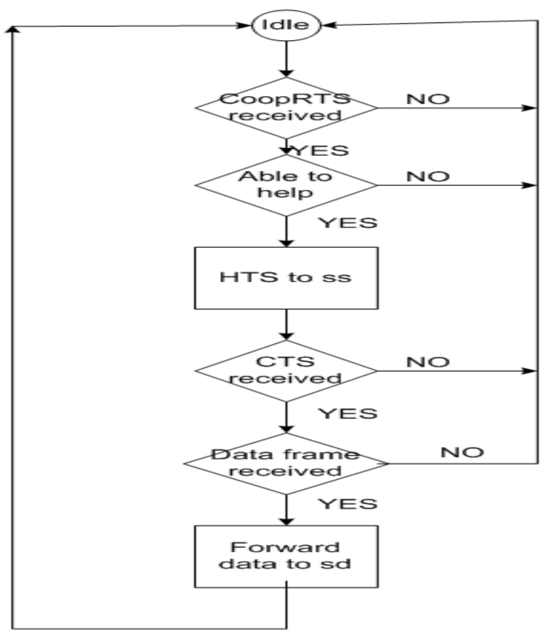



Figure 3.8: FLow Chart for Helper Station

#### 3.5.2.2. Flow chart at Desination station

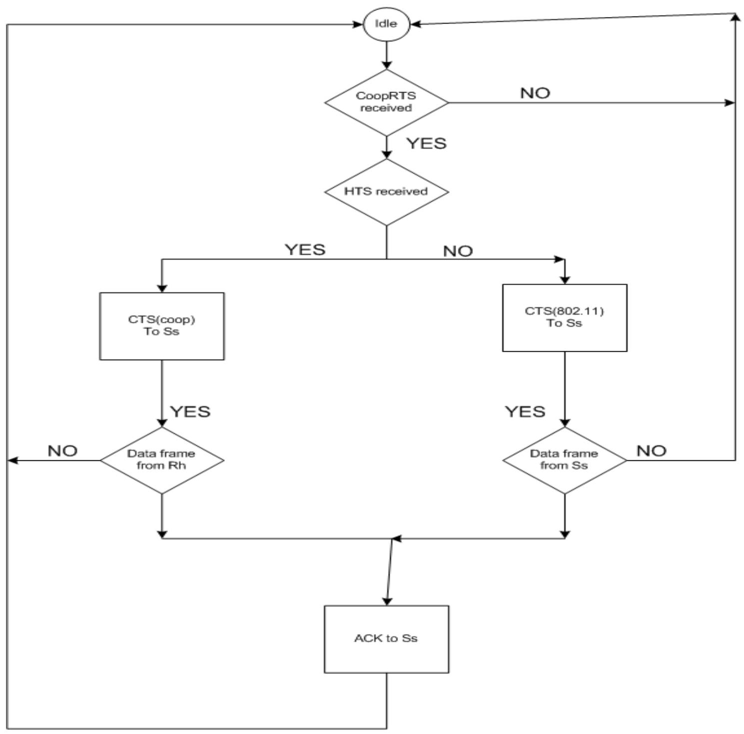



Figure 3.9: FLow Chart for Destination Station

- If $S_d$ receives a CoopRTS, whose RA field is set to the MAC address of $S_d$, $S_d$ should wait for the corresponding HTS message from $S_h$.

- If $S_d$ hears an HTS message from $S_h$, it transmits a CTS message back to $S_s$ after a SIFS interval, with the duration field.If the data frame does not arrive before this timer expires, it assumes that the data transmission was aborted and goes back to the initial state.

- If the HTS message is not received by $S_d$ after a SIFS interval, $S_d$ follows the standard 802.11 approach and transmits the CTS message after another SIFS interval, with the duration field.

## 3.2 summary

In this chapter, The CoopMAC protocol is described in detail. This protocol is compatible with IEEE 802.11 legacy. The frame formats of various handshaking signals are also described and flow charts at source, destination, helper stations.

# Chapter 4

# Simulation Results for IEEE 802.11(WLAN)

The CoopMAC protocol is based on the IEEE 802.11 DCF mechanism. In this chapter, We have first done the simulation for IEEE 802.11 using different routing protocol such as AODV, DSDV, DSR etc. with different network conditions. The simulation results are carried out using custom event driven simulator - Network Simulator-2 (NS-2)(Appendix A).

## 4.1 Overview of Different Routing Protocol

In recent years many protocols , for ad hoc wireless Networks have been developed which are Distance sequence distance vector (DSDV), Ad hoc on demand distance vector (AODV), Dynamic source routing (DSR)[19] etc. There are two main categories of routing protocols for ad hoc wireless networks which are:

- Table driven routing protocols (Proactive)
- On demand routing protocols (Reactive)

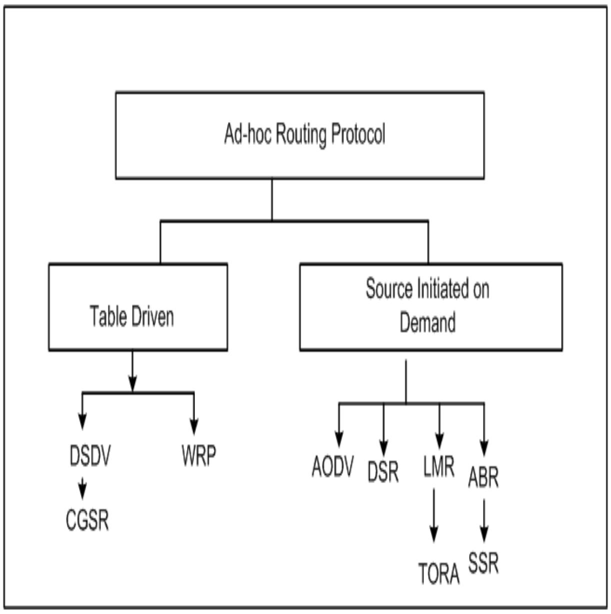



Figure 4.1: Categorization of ad hoc routing protocols

### 4.1.1 Table driven routing protocols (Proactive)

These protocols are also called as proactive protocols since they maintain the routing information even before it is needed. Each and every node in the network maintains routing information to every other node in the network. Routes information is generally kept in the routing tables and is periodically updated as the network topology changes. Many of these routing protocols come from the link-state routing. There exist some differences between the protocols that come under this category depending on the routing information being updated in each routing table. Furthermore, these routing protocols maintain different number of tables. The proactive protocols are not suitable for larger networks, as they need to maintain node entries for each and every node in the routing table of every node. This causes more overhead in the routing table leading to consumption of more bandwidth.[19]

### 4.1.2 On demand routing protocols (Reactive)

These protocols are also called reactive protocols since they don't maintain routing information or routing activity at the network nodes if there is no communication. If a node wants to send a packet to another node then this protocol searches for the route in an on-demand manner and establishes the connection in order to transmit and receive the packet. The route discovery usually occurs by flooding the route request packets throughout the network.

## 4.2 Ad hoc On Demand Distance Vector Routing (AODV) protocol

Ad hoc On-Demand Distance Vector (AODV) routing is a routing protocol for mobile ad hoc networks and other wireless ad-hoc networks. It is an on-demand and distance-vector routing protocol, meaning that a route is established by AODV from a destination only on demand.[20]

AODV is capable of both unicast and multicast routing. It keeps these routes as long as they are desirable by the sources. Additionally, AODV creates trees which connect multicast group members. The trees are composed of the group members and the nodes needed to connect the members. The sequence numbers are used by AODV to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of mobile nodes.

### 4.2.1 Destination Sequenced Distance Vector (DSDV) Protocol

The destination sequenced distance vector routing protocol is a proactive routing protocol. This protocol adds a new attribute, sequence number, to each route table entry at each node. Routing table is maintained at each node and with this table; node transmits the packets to other nodes in the network. This protocol was motivated for the use of data exchange along changing and arbitrary paths of interconnection which may not be close to any base station.

Each node in the network maintains routing table for the transmission of the packets and also for the connectivity to different stations in the network. These stations list for all the available destinations, and the number of hops required to reach each destination in the routing table. The routing entry is tagged with a sequence number which is originated by the destination station. In order to maintain the consistency, each station transmits and updates its routing table periodically. The packets being broadcasted between stations indicate which stations are accessible and how many hops are required to reach that particular station. The packets may be transmitted containing the layer 2 or layer 3 address.

Routing information is advertised by broadcasting or multicasting the packets which are transmitted periodically as when the nodes move within the network. The DSDV protocol requires that each mobile station in the network must constantly; advertise to each of its neighbors, its own routing table. Since, the entries in the table may change very quickly; the advertisement should be made frequently to ensure that every node can locate its neighbors in the network. This agreement is placed, to ensure the shortest number of hops for a route to a destination; in this way the node can exchange its data even if there is no direct communication link.

### 4.2.2 Dynamic Source Routing (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. Using DSR, the network is completely self-organizing and self-configuring, requiring no existing network infrastructure or administration. Network nodes 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 anytime, the resulting network topology may be quite rich and rapidly changing.

Dynamic Source Routing (DSR) is a routing protocol for wireless mesh networks. It is similar to AODV in that it establishes a route on-demand when a transmitting mobile node requests one. However, it uses source routing instead of relying on the routing table at each intermediate device.

Dynamic source routing protocol (DSR) is an on-demand, source routing protocol, whereby all the routing information is maintained (continually updated) at mobile nodes. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. The protocol is composed of the two main mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network .

## 4.3 Performance Results for IEEE 802.11(WLAN)

### 4.3.1 Simulation with Fixed Node position

Figure 4.2 shows the Simulation scenario with 5 nodes. In this scenario, the position of the nodes are fixed manually using two-dimensional co-ordinate system in the flat grid area of 300X300. Thus, at the starting of the simulation, node positions are fixed. As the simulation runs, the nodes move towards each others. This movement between the nodes has been decided sing coordinates i.e. which nodes moves towards which node at which time or going away from which node at which time etc. Then after TCP traffic has been assigned between different nodes. Here, the data rate between the source and destination is 1Mbps.

This simulation are carried out to determine the throughput for different routing protocol. Figure 4.3 shows the throughput for AODV routing protocol and Figure **??** shows the throughput result for DSR routing protocol. (Throughput means how many packets

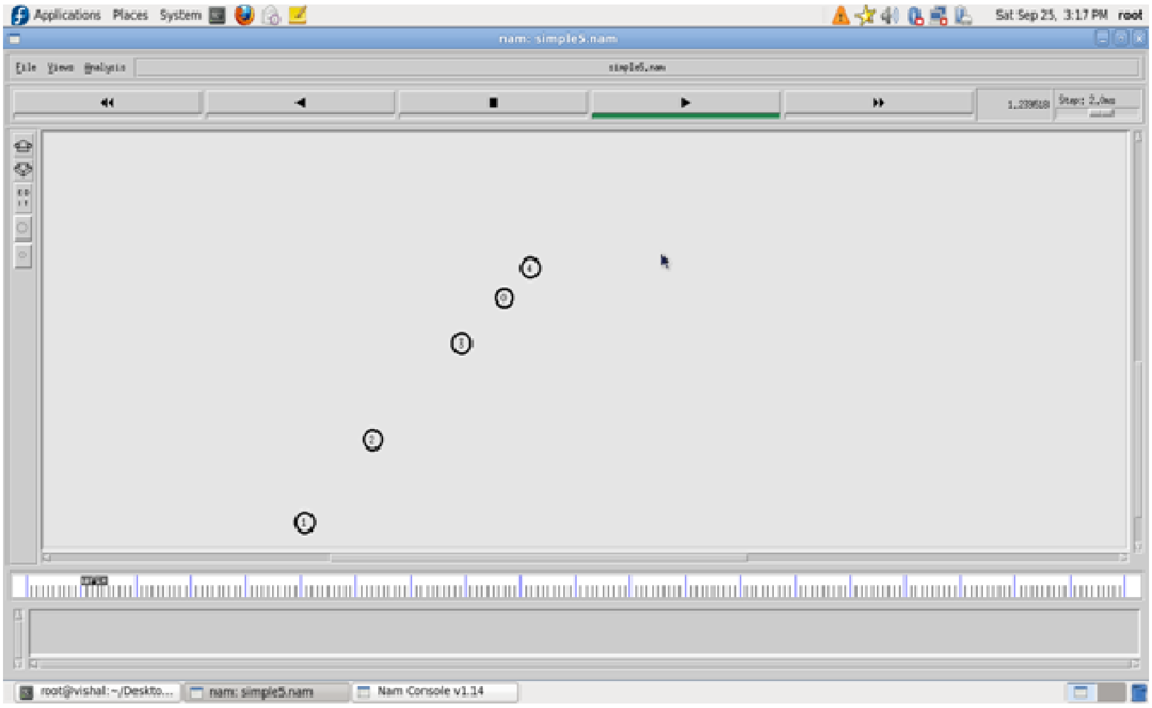



Figure 4.2: Simulation Scenario-I

are successfully received by the destination out of total packet sent by the source).

From the results, it is seen that, the throughput is decreases as the no. of nodes is increases. The rate of change of throughput as the no of nodes increases, is higher in DSR protocol than that of AODV protocol. As the no of nodes increases the load on each node is also increases and hence the throughput decreases.

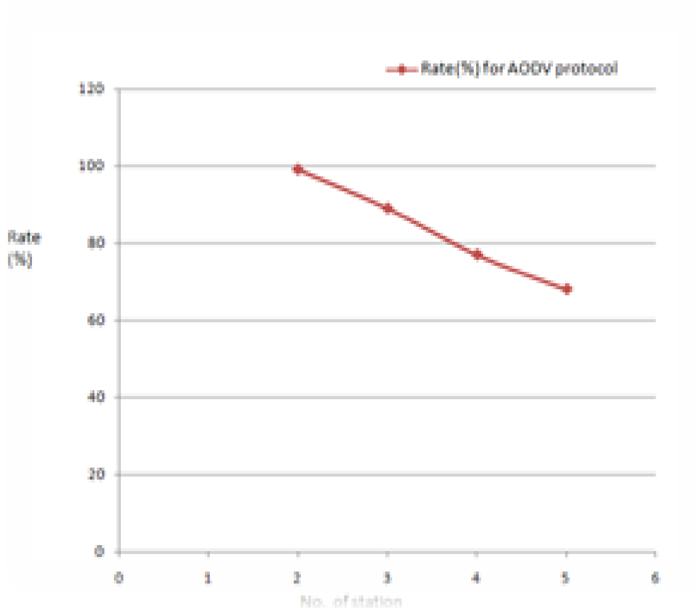



Figure 4.3: Throughput For AODV

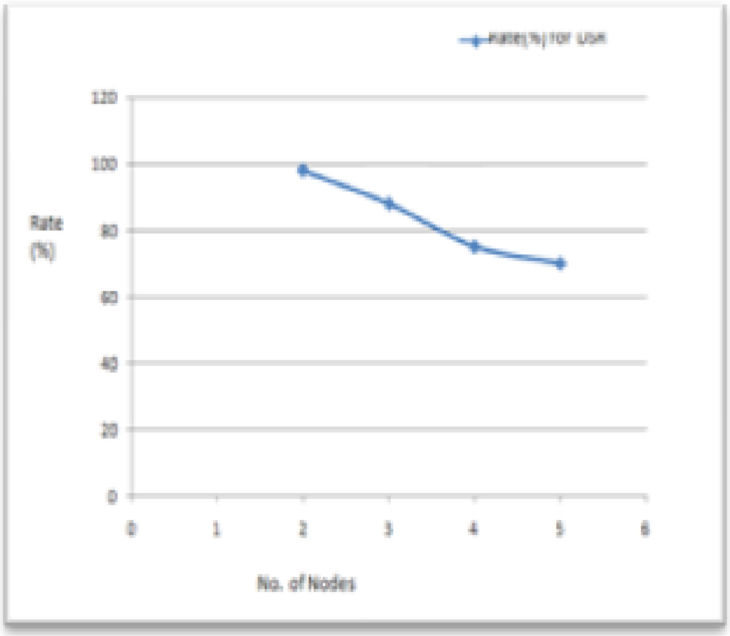



Figure 4.4: Throughput For DSR

### 4.3.2 Simulation with Random Node Movement

Figure 4.5 shows the Simulation scenario with 20 nodes. In this scenario, we have fixed the position of the nodes randomly using two-dimensional co-ordinate system in the flat-grid area of 300X300. Thus, at the starting of the simulation, node position are random in the flatgrid area of 300X300. The node movements is also random in manner. As the simulation runs, the nodes moves towards each other. This movement between the nodes has been decided using coordinates i.e. which nodes moves towards which node at which time or going away from which node at which time etc. Then after TCP traffic has been assigned between different nodes. Here, the data rate between the source and destination is 1Mbps.

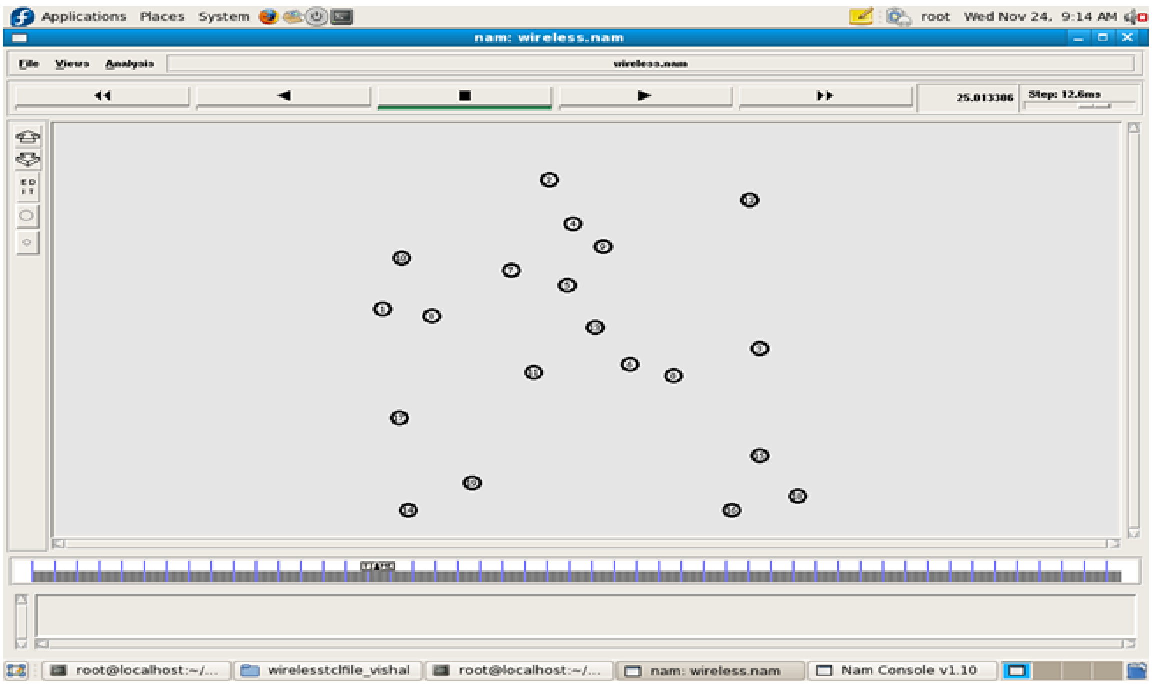



Figure 4.5: Simulation scenario-II

The simulations are carried out no of times. These simulation are carried out to determine the throughput for different routing protocol. Figure 4.6, Figure 4.7 and Figure 4.8 shows the throughput for AODV, DSDV, DSR routing protocols respectively.

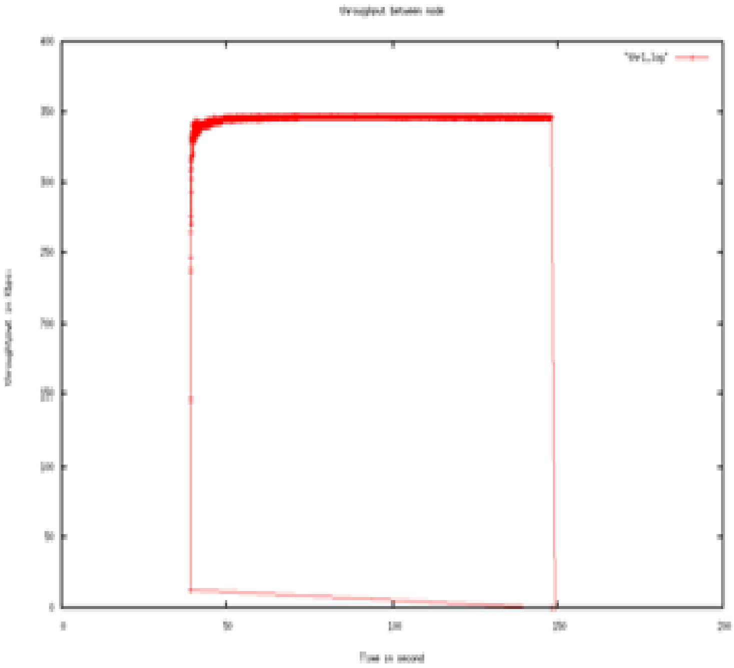


Figure 4.6: Throughput For AODV

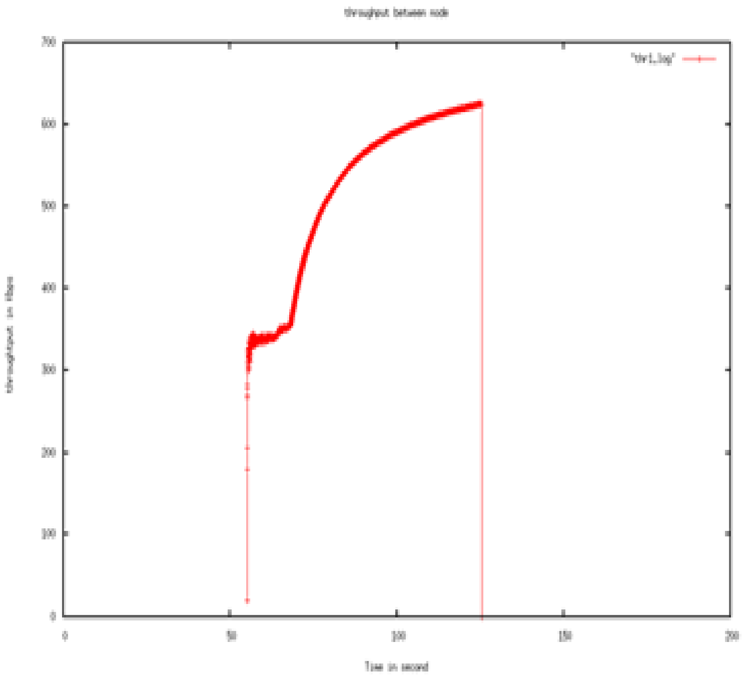


Figure 4.7: Throughput For DSDV

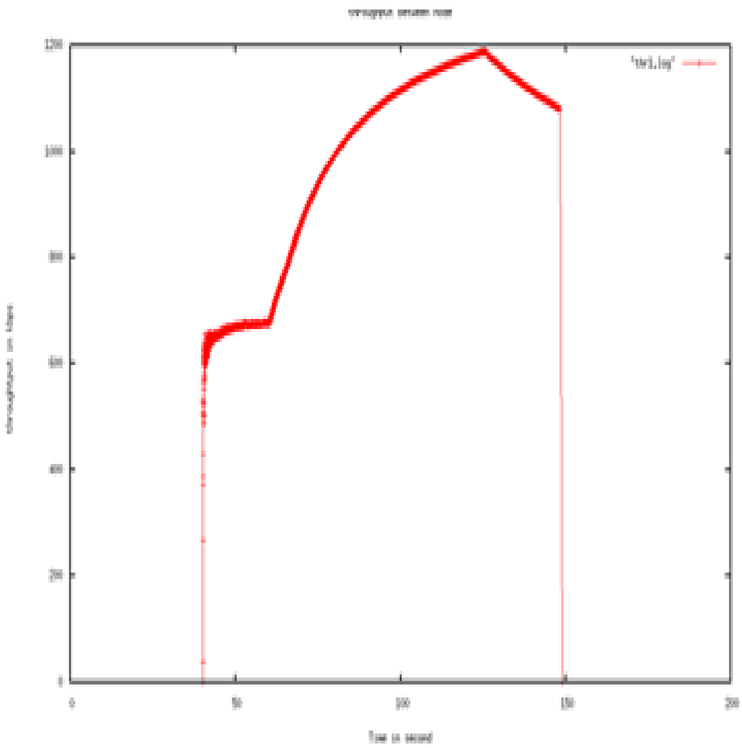



Figure 4.8: Throughput for DSR

From the above results, it is seen that for AODV protocol, the throughput is almost constant during all time. In the DSDV protocol, the throughput is increases as time increases and in DSR protocol the throughput is decreases after some time. The reason behind this change in results is due to the difference in routhing mechanism of each protocol.

## 4.4 summary

In this chapter, various Routing protocol are described.The results of Simulation for IEEE 802.11 with different Routing Protocol with fixed node position and random node position are also presented.

# Chapter 5

# Simulation Results for CooopMAC Protocol

In previous chapter 4, we have discussed about IEEE 802.11 and presented the results of simulations carried out. In this chapter, we will perform various simulations for cross-layer protocol, CoopMAC, which is compatible with IEEE 802.11 legacy.

One of the aim of the CoopMAC protocol is that it allows the high rate stations to help stations that can only sustain a low data rate to mitigate this effect. However, modification is needed to existing MAC protocols, like 802.11 DCF mechanism, to enable such cooperation. The Cross-layer protocol, CoopMAC is based on IEEE 802.11 DCF mechanism.

For Simulation of CoopMAC protocol, we have modified the custom event-driven simulator - Network Simulaotor-2. In this custom event-driven simulator - Network Simulator-2, we have mainly modified the header file mac_80211.h and mac_80211.cc files to enable cooperation at mac layer. It generates events faithfully following every state transition of the 802.11 MAC including the head of line arrival, back off count down, DIFS time, individual transmissions, RTS/CTS transmissions for both legacy and CoopMAC protocol. As shown in Table 5.1 and in Table 5.2, the set of core parameters used in the simulation assume the default values of specified in 802.11b standard.

Table 5.1: Parameters used in Simulation

| | |
|---|---|
| MAC header | 272 bits |
| PHY header | 192 bits |
| RTS | 352 bits |
| CTS | 304 bits |
| ACK | 304 bits |
| Data rate for MAC and PHY header | 1 Mbps |
| Slot time | 20$mu$s |
| SIFS | 10 $mu$s |
| DIFS | 50 $mu$s |
| aCWmin | 31 slots |
| aCWmax | 1023 slots |
| retryLimit | 6 |

Table 5.2: MODE TABLE (PATH LOSS EXPONENT (PLE) = 3)

| Data Rate | 11 Mbps | 5.5 Mbps | 2 Mbps | 1 Mbps |
|---|---|---|---|---|
| Range | 48.2 m | 67.1 m | 74.7 m | 100 m |

We study the performance of each approach based on the protocol aspects as listed below:

- Throughput performance
- Delay performance (e.g., average end-to-end delay and transmission delay)
- Impact of cooperation on the helper station that helps the source transmit to the destination.

## 5.1 Simulation Scenario for CoopMAC and Performance Results

Fig 5.1 shows the simulation scenario for CoopMAC protocol with three nodes, in which one node act as a source, second node act as a destination and third node act as a helper. In this Simulation, the data rate for the simulation are as shown in the figure **??**.

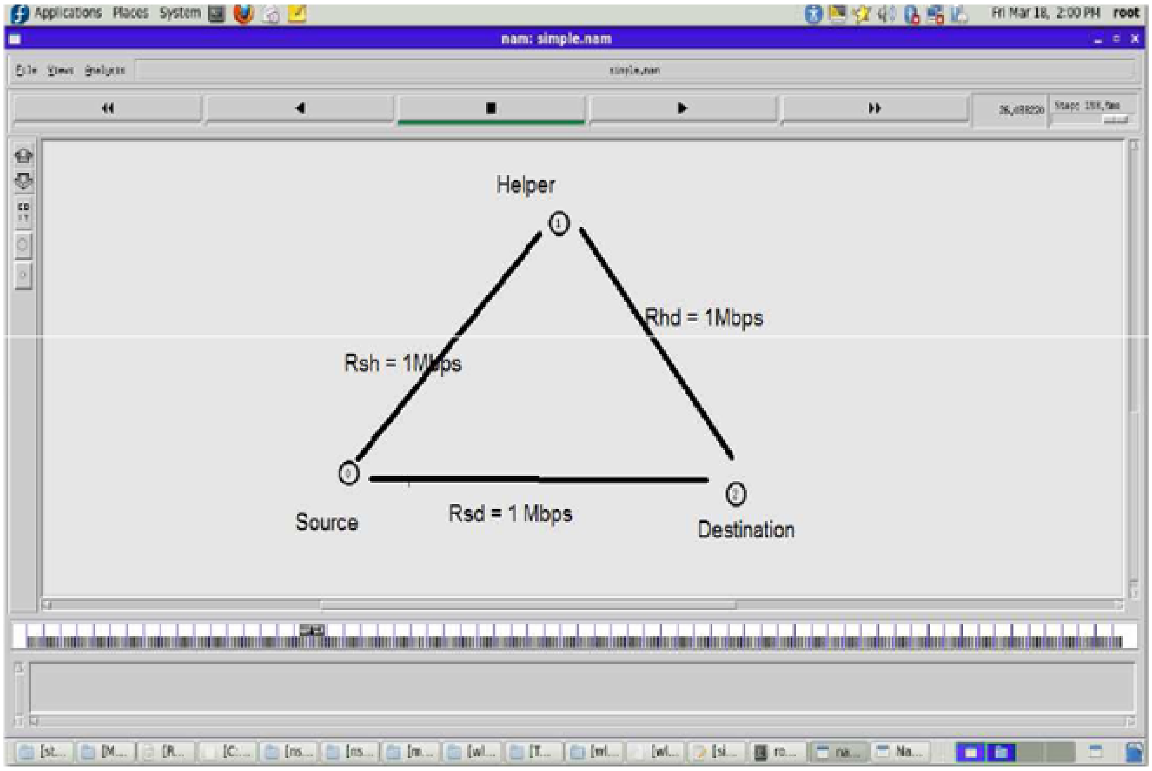



Figure 5.1: Simulation Scenario

Now first we assigned the TCP traffic between the source and destination and the throughput is calculated for both CoopMAC protocol and its legacy IEEE 802.11 as shown in Figure 5.2. In second simulation, we assigned the UDP traffic between the source and destination and then calculate the throughput for both CoopMAC protocol and its legacy IEEE 802.11, as shown in figure 5.3.

The increase in throughput for CoopMAC is due to cooperation enabled between these three nodes. As the source forwards its data to destination as well as to the helper. Now as the distance between the helper and destination is lower than that of direct path. So throughput is more for coopMAC protocol as compared to its legacy IEEE 802.11 standards for both TCP(Transmission Control Protocol) as well as UDP(User Datagram Protocol).

As seen from the results, the throughput for UDP is higher than that of TCP. The reason for this is that TCP is connection-oriented services i.e. in TCP, an ack signal is

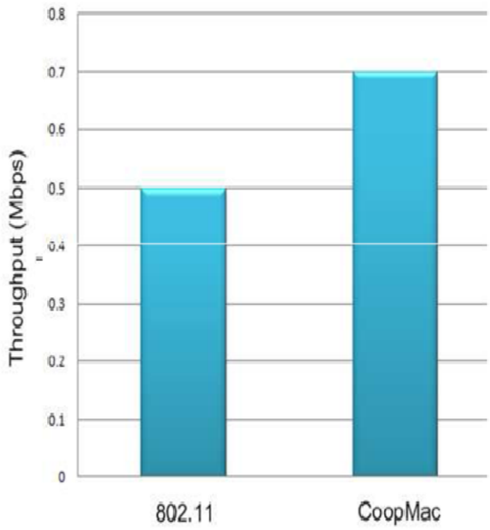



Figure 5.2: Throughput for TCP

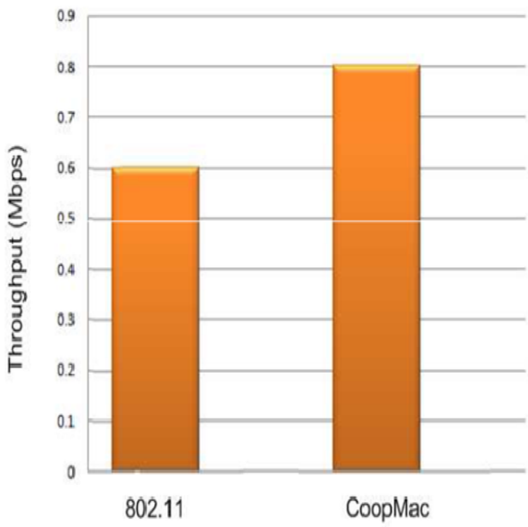



Figure 5.3: Throughput for UDP

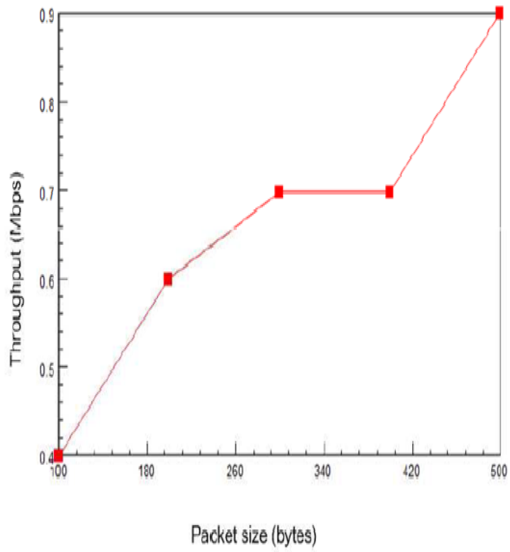



Figure 5.4: Throughput V/s Packet Size for CoopMAC

required while in UDP, an ack signal is not required.

The Fig 5.4 shows the results for Throughput V/s Packet size. As seen from the figure 5.4, the throughput increases as the packet size increases. As the packet size increases, total no of packets is decreases and hence no of drop-out in packets is also decreases and hence the increase in the throughput. The maximum packet length of MPDU is of 8192 bytes.

Fig 5.5, fig 5.6 and fig 5.7 shows the comparison of throughput for both IEEE 802.11 and CoopMAC protocol with different packet length of 512 bytes, 1024 bytes and 2048 bytes respectively. As the packet length is increases the total no of packets is decreases and hence increase in throughput.

Now, figure 5.8 shows the simulation results for delay for both 802.11 and CoopMAC protocol. The delay for CoopMAC protocol is lower than that of IEEE 802.11. Because as IEEE 802.11 provides the multirate capability and CoopMAC protocol is compatible with it, it also provides multirate capability. In coopMAC protocol, helper station forward

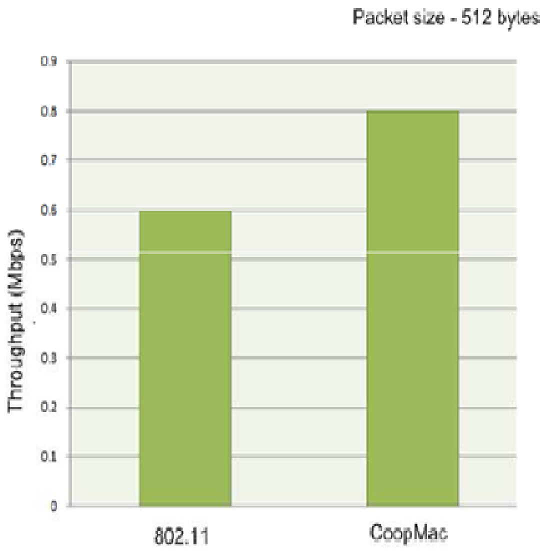



Figure 5.5: Throughput V/s Packet size

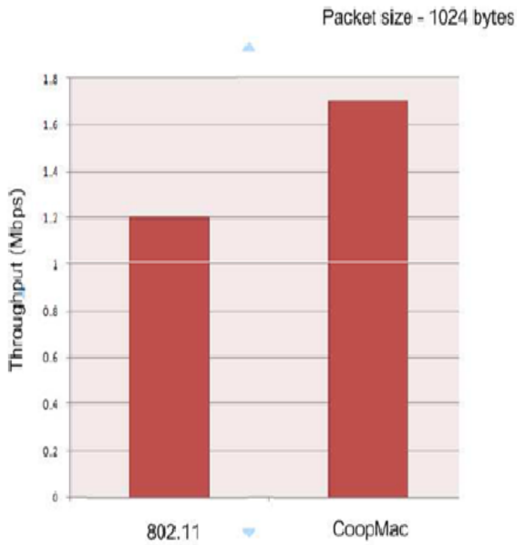



Figure 5.6: Throughput V/s Packet size

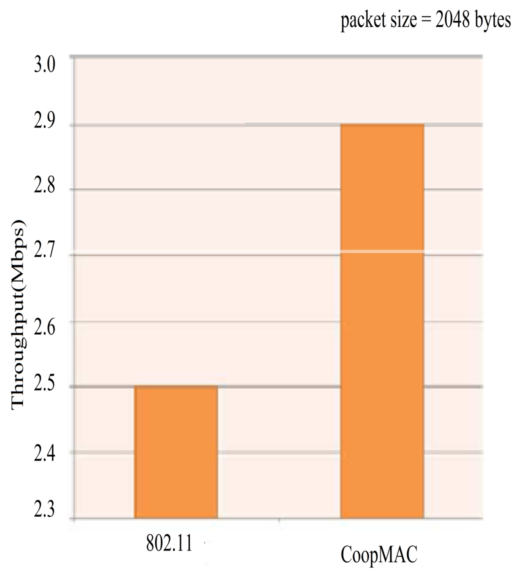



Figure 5.7: Throughput V/s Packet size

source station data to the destination and distance between helper and destination is small as compared to direct distance between the source and the destination and hence higher rate and lower delay than that of its legacy.

Now, We change the data rate between source, destination and helper as shown in the table 5.3and then calculate the throughput for the same.

Table 5.3: Different Data rate for CoopMAC

| **Sr N.** | **Rsd(Mbps)** | **Rsh(Mbps)** | **Rhd(Mbps)** |
|---|---|---|---|
| 1 | 1 | - | - |
| 2 | 1 | - | - |
| 3 | 1 | 1 | 1 |
| 4 | 1 | 2 | 5.5 |
| 5 | 1 | 5.5 | 5.5 |
| 6 | 1 | 5.5 | 11 |
| 7 | 1 | 11 | 11 |

As seen from the result fig. 5.9, the throughput for different data rates is increases, as

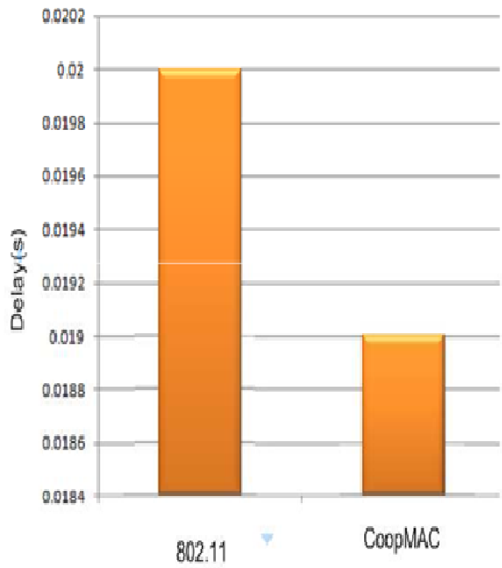



Figure 5.8: delay for 802.11 and CoopMAC

datarate increases. For higher throughput both data rate must be same and high.

Table 5.4 shows datarate setting for study of end to end delay between the stations. Figure 5.10 shows result of end to end delay for different packet rate for both 802.11 and coopMAC protocol. As the packet rate increases, more no of packets has to be sent by the helper or source to the destination and hence it increases the delay. The delay for coopMAC protocol is 50% lower than that of 802.11 as seen in figure 5.10.

Table 5.4: setting for study of end to end delay

| **Sr N.** | **Rsd(Mbps)** | **Rsh(Mbps)** | **Rhd(Mbps)** |
|---|---|---|---|
| 1 | 1 | - | - |
| 2 | 1 | 2 | 5.5 |
| 3 | 1 | 5.5 | 5.5 |
| 4 | 1 | 11 | 5.5 |

Figure 5.11 shows the delay for different packet sizes. The delay for CoopMAC with (11-5.5Mbps rate) is the lowest in all the results shown in the figure 5.11.

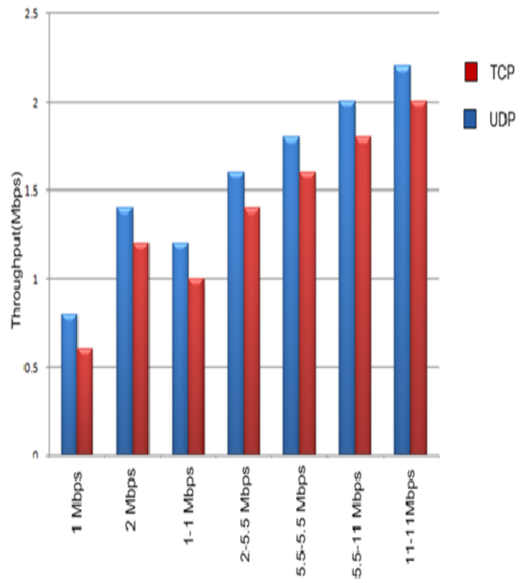



Figure 5.9: throughput for different data rates

Figure 5.12 shows the simulation results for throughput V/s simulation time for 802.11 and stationary coopMAC and CoopMAC with mobility.

Fig 5.13 shows the result of transfer time required for sending packets from source to destination with 802.11 for direct link of 2Mbps and with CoopMAC protocol. The CoopMAC with rate 11-11Mbps performs better than all others as shown in figure. The transfer time for CoopMAC with (11-11 Mbps rate) is about 50% less than that of direct 802.11 with 2Mbps due to cooperation between the nodes.

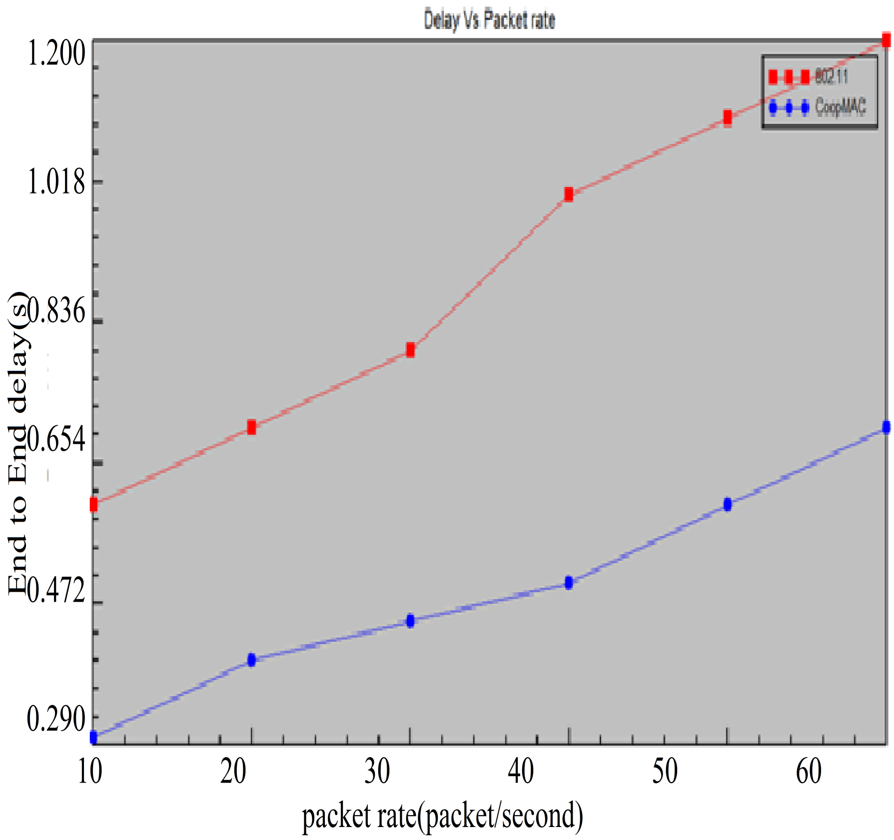



Figure 5.10: End to end delay V/s packet rate(packet/second)

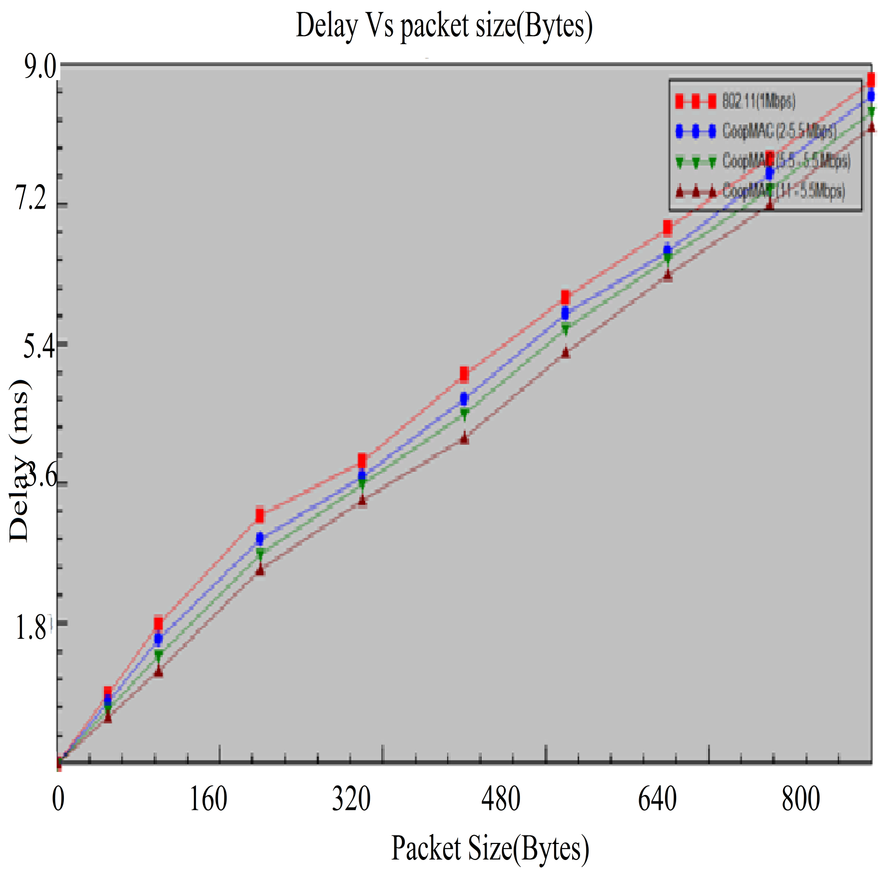



Figure 5.11: End to end delay V/s packet size(Bytes)

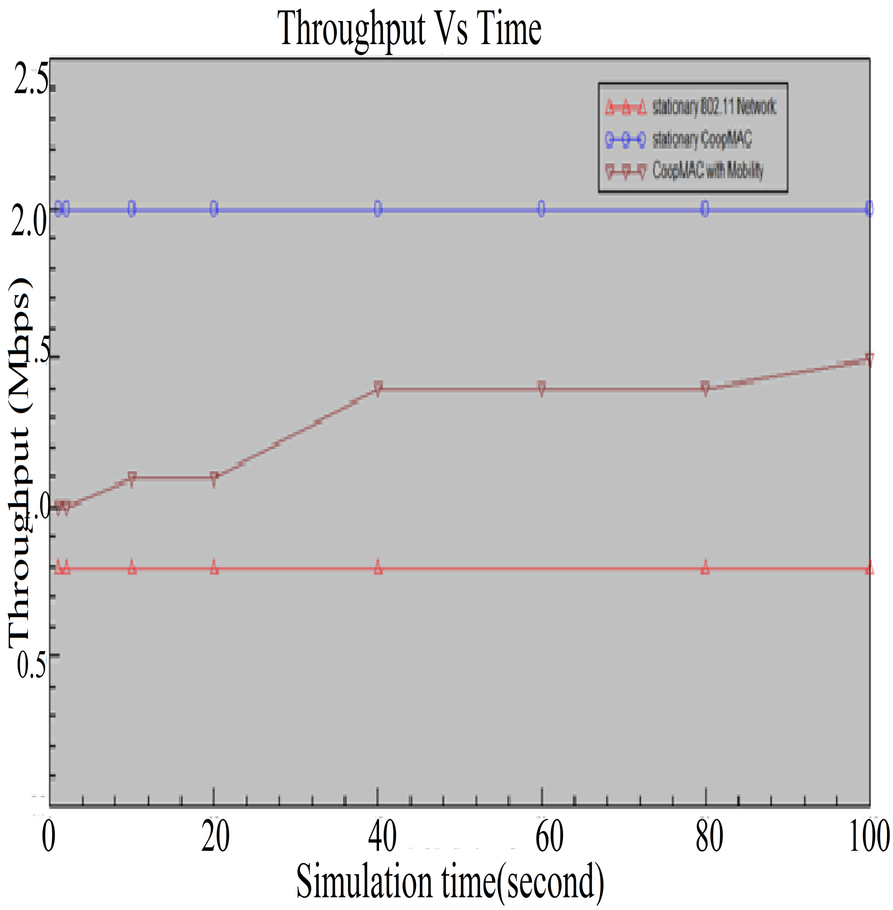



Figure 5.12: Throughput V/s Simulation time(second)

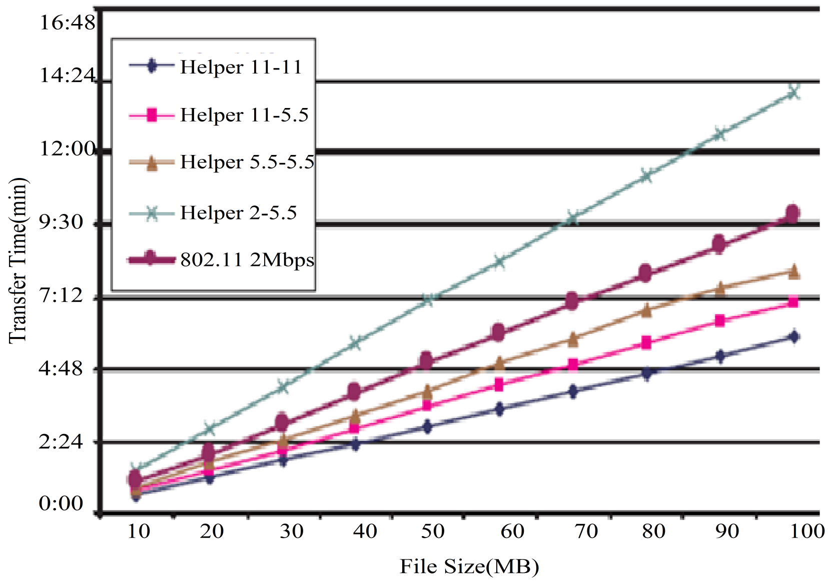



Figure 5.13: CoopMAC Vs 802.11 with a 2Mbps direct link

## 5.2 summary

In this chapter, The various results (throughput,delay) of Simulations with three nodes for CoopMAC protocol are presented and discussed. The comparison between various results of IEEE 802.11 and CoopMAC are also presented.

# Chapter 6

# Conclusion and Future work

## 6.1 Conclusion

The aim of this dissertation is to explore the cooperation at the medium access control (MAC) layer and do the performance analysis of a protocol called CoopMAC, which is based upon the existing IEEE 802.11 DCF mode. The various simulation has been done for IEEE 802.11 for throughput and delay with different network scenario. We have developed custom event-driven simulator for CoopMAC protocol. The various simulation has been done for CoopMAC protocol while keeping the number of nodes limited to 3. At last, the comparison between the IEEE 802.11 and CoopMAC protocol is presented. From comparison it is concluded that,

- The throughput for CoopMAC protocol is higher than its legacy IEEE 802.11. The reason behind this is incorporation of cooperation at the mac layer.
- The End-to-end delay from source to destination is decreases as data rate between Source to helper and helper to destination is higher than the direct transmission.
- The cooperation at mac layer enables high data rate stations to assist low data rate stations to help them in forwarding their data to the destination.
- The data rate between Source to helper and helper to destination should be high and same.

## 6.2 Future scope

In Future, Cooperation can be implemented at the Medium access control layer with more no. of nodes and can be done various performance analysis of it and comparison between CoopMAC protocol and IEEE 802.11 protocol.

The performance of this CoopMAC protocol can be implemented using Software Defined Radio or Driver Approach(HostAP.) for real environment performances.

# Appendix A

# Introduction to Network Simulator-2

Network Simulator, popularly known as NS-2, is a discrete event network simulator in which each event occurs at an instant in time. NS is an object oriented simulator, written in C++, with an OTcl interpreter as a frontend. It is used to simulate routing and multicast protocols. It supports large number of network protocols for simulation and provides results for wired, wireless and wired-cum-wireless scenarios. NAM (Network Animator) is the most important feature of NS. It gives the visual outputs of the simulation scenarios. As NS is open source and plentiful documentation is available, it is very popular for simulations and extensions[18].

Thus Network Simulator-2 is a...

- Discrete Event Simulator
- Packet level Simulator
- used for Modeling Network protocols

- Collection of Various protocols at multiple layers
  - TCP(reno, tahoe, vegas, sack)
  - MAC(802.11, 802.3, TDMA)
  - Ad-hoc Routing (DSDV, DSR, AODV, TORA)
  - Sensor Network (diffusion, gaf)
  - Multicast protocols, Satellite protocols, and many others

## A.1 NS-2 Architecture

NS uses two different languages because the simulator has two different kinds of things it needs to do. First, detailed simulation of protocol requires system programming language which can efficiently manipulate bytes, packet headers and implement algorithms that run over large data sets. For these tasks, run-time speed is important and turn-around time (run simulation, find bug, fix bug, recompile, re-run) is less important. On the other hand, a large part of network research involves slightly varying parameters or configurations or quickly exploring a number of scenarios. In these cases, iteration time (change the model and re-run) is more important. Since configuration runs once (at the beginning of the simulation), run-time of this part of the task is less important[18].

NS meets both of these needs with two languages, C++ and OTcl. C++ is fast to run but slower to change, making it suitable for detailed protocol implementation. OTcl runs much slower but can be changed very quickly (and interactively), making it ideal for simulation configuration. NS (via tclcl) provides glue to make objects and variables appear on both languages.

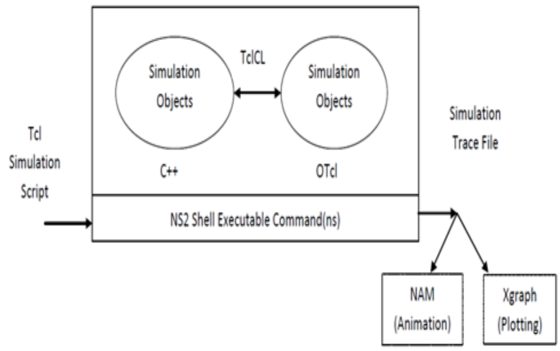



Figure A.1: NS-2 Architecture

## A.2 Basic of NS2

The NS covers large number of applications, of protocols, of network types, of network elements and of traffic models.

- NS is based on two languages: an object oriented simulator, written in C++,and a OTcl (an object oriented extension of TCL) interpreter, used to excuse users command scripts.
- 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 in particular useful for the detailed definition of operation of protocols. This allows one to reduce packet and event processing time.
- Then in the OTcl script provided by the user, we can define a particular network topology, the specific protocols and applications that we wish to simulate (whose behavior is already defined in the compiled hierarchy) and the form of the output that we wish to obtain from the simulator. The OTcl can make use of the objects

compiled in C++ through an OTcl linkage (done using tclcl) that creates a matching of OTcl object for each of the C++.

- NS is a discrete event simulator, where the advance of time depends on the timing of events which are maintained by scheduler. An event is an object in the 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 (there are four, but by default NS use a simple linked list) with events to be executed and fires them one by one, invoking the handler of the event.

## A.3 TCL and OTcl programming

TCL (Tool Command Language) is a language with a very simple syntax and it allows very easy integration with other languages. TCL was created by John Ousterhout. The characteristics of this language are:

- It allows fast development
- It provides graphic interface
- It is compatible with many platforms
- It is flexible for integration
- It is easy to use
- It is free

## A.4 Contents of different file

**TCL file,**

- First we set the variables according to our requirements like radio propagation model, area, channel type, MAC type, interface queue type, link layer type, number of mobile nodes, time of simulation, energy model.

- Second we set trace file and nam file in which we want to put our results.
- Third we set up topography of network and configuration of node.
- Fourth we set movement model which defines movement of nodes.
- Fifth we set traffic model which describes traffic pattern between nodes.

**Awk file,**

Awk is one type of scripting language. Awk file use for extracting useful data from trace file which is generated by simulation of Tcl file. From this useful data we can conclude about the behavior of network.

**In gnuplot,**

Gnuplot is used to plot graph of in 2-D. Gnuplot we have to create .dem file in which parameters like x title, y title, color etc.

# A.5 Installation of NS2

## A.5.1 Installation of Fedora Core 12

The various platforms supported for NS are Linux, Solaris and Windows XP using Cygwin. We have used Fedora core 12 (Linux environment). Following are the steps to install Fedora core 12 OS. Fedora core 8 is freely available.

- Remove data from one of the drives where fedora is to be installed.
- Load the CD in drive.
- Select GUI mode by pressing enter.
- Skip for disc checking.
- Then select the English US language.

- Select Custom layout for partitioning hard disk and click next.
- Where we want to install, make that disk space free first (at least 5GB). The drives are named as devsda1 (which is C drive in the system) and so on for the other drives.
- Give a mount point as (root) & select ext3.
- Then make it ok.
- Select grub boot loader & make windows or other OS as by default.
- Select all the software if needed.
- Give the computer name
- Select region as AsiaCalcutta.
- Enter Root password
- Then installation will start (1103 packet should be installed for proper functioning of fedora).
- Now The Fedora core 12 has been installed in your system.

### A.5.2 Installation of NS-2 (ns-2.34)

The installation procedure of NS-2 in Fedora core 8 OS is as follows. It should be installed in the root account of the fedora. NS-2 is freely available.

- Go to link: SourceForge.net: Files.
- Download file: ns-allinone-2.34.tar.gz
- Open the root account in Fedora.
- Extract the downloaded file on desktop.
- Open console and give following commands.
- cd Desktop

- cd ns-allinone-2.34
- ./install
- Now, the installation will start and after some time the message named "IMPORTANT NOTICES" will appear on screen which gives the information about the paths to be added.

(1) You MUST put /home /myusername /ns-allinone-2.24 /otcl-1.11, /home / myusername /ns-allinone-2.24 /lib, into your LD_LIBRARY_PATH
environment variable.

If it complains about X libraries, add path to your X libraries into LD_LIBRARY_PATH. If you are using csh, you can set it like: setenv LD_LIBRARY_PATH <paths> If you are using sh, you can set it like: export LD_LIBRARY_PATH=<paths>
(2) You MUST put /home/myusername/ns-allinone-2.34/tcl8.4.11/library into your TCL_LIBRARY environmental variable. Otherwise ns/nam will complain during startup.
(3) [OPTIONAL] To save disk space, you can now delete directories tcl8.4.11 and tk8.4.11. They are now installed under /home/myusername/ns-allinone-2.34 /bin,include,lib
After these steps, you can now run the ns validation suite with cd ns-2.34; ./validate
For trouble shooting, please first read ns problems page http://www.isi.edu/nsnam/ns/ns-problems.html. Also search the ns mailing list archive for related posts.

- Give the following command in already opened console.
  - gedit .bashrc
- Then the "gedit" text editor will open which is the official text editor for the GNOME (GNU Object Model Environment) desktop.
- Keep the written things in that editor as it is and add the following lines at the end of it. Replace /usr/local/" by installation path like "/root/Desktop". Also, make changes according to the version of the ns. For example, for ns-2.34, otcl?s version is 1.14, tcl?s version is 8.4.14, etc.

(i) #LDLIBRARY_PATH

OTCL_LIB=/usr/local/ns-allinone-2.34/otcl-1.14
NS2_LIB=/usr/local/ns-allinone-2.34/lib
X11_LIB=/usr/X11R6/lib
USR_LOCAL_LIB=/usr/local/lib
export LD_LIBRARY_PATH = $LD_LIBRARY_PATH: $OTCL_LIB: $NS2_LIB:
$X11_LIB: $USR_LOCAL_LIB
# TCL_LIBRARY TCL_LIB=/usr/local/ns-allinone-2.34/tcl8.4.14/library
USR_LIB=/usr/lib
export TCL_LIBRARY=$TCL_LIB:$USR_LIB
#PATH
XGRAPH=/usr/local/ns-allinone-2.34/bin:/usr/local/ns-allinone-
2.34/tcl8.4.14/unix:/usr/local/ns-allinone-2.34/tk8.4.14/unix
NS=/usr/local/ns-allinone-2.34/ns-2.34/ NAM=/usr/local/ns-allinone-2.34/nam-1.14/
PATH=$PATH:$XGRAPH:$NS:$NAM

- Now, give the following command in the already opened console.
  - source .bashrc
- Reopen the console now.
- Write ns and press enter. If there is no error, a "%" will appear on screen.
- Press exit to quit this mode or back to prompt

## A.6 Steps for Simulation in NS-2

Following are the steps to write tcl file to in NS which is necessary to run the simulation.

- Define goal and expected result

- Create network topology
- Nodes Link (simplex/duplex, BW, delay, etc.).
- Specify agents (TCP, UDP, etc.)
- Traffic sources (CBR).
- Simulation scenario (wired, wireless, wired-cum-wireless)

# References


[1] Yan Zhang, Hsiao-Hwa Chen , Mohsen Guizani, *"Cooperative Wireless Communications"*,CRC Press.

[2] Aria Nosratinia, Todd E. Hunter, Ahmadreza Hedaya, Cooperative Communication in Wireless Networks, *IEEE Communications Magazine*,October 2004,pp.74-80.

[3] A. Sendonaris, E. Erkip, and B. Aazhang, User Cooperation Diversity Part I and Part II,, *IEEE Trans. Commun.*, vol. 51, no. 11, Nov. 2003, pp. 192748.

[4] A. Sendonaris, E. Erkip, and B. Aazhang, User Cooperation Diversity Part II: Implementation Aspects and Performance Analysis, *IEEE Trans. Commun.*, vol. 51, Nov.2003, pp. 193948

[5] G. Kramer, I. Maric, and R. D. Yates,*"Cooperative Communications"*, Hanover, MA, USA, 2006.

[6] A. Sendonaris, E. Erkip, and B. Aazhang,"User Cooperation Diversity Part I: System Description," *IEEE Trans. Commun.*, vol. 51, Nov. 2003, pp. 1927-38

[7] J. N. Laneman, D. Tse, and G. W. Wornell, "Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior", *IEEE Trans. Info. Theory*, vol. 50, Dec. 2004, pp. 3062-80.

[8] Lun Dong, *"Cross-Layer Design For Cooperative Wireless Networks"*,Ph.D. Thesis, Drexel University.

[9] T. M. Cover and A. A. E. Gamal,"Capacity Theorems for the Relay Channel", *IEEE Trans. Info. Theory*, vol. 25, no. 5, Sept. 1979, pp. 572-84.

[10] P. Liu, Z. Tao, Z. Lin, E. Erik, and S. Pan war, "Cooperative wireless communications: A cross-layer approach", *IEEE Wireless Communications*, vol. 13, no. 4, pp. 84-92, 2006.

[11] P. Liu, Z. Tao, S. Narayanan, T. Koraris, and S. S. Panwar, "Coop MAC: a cooperative MAC for wireless LANs",*IEEE J. Selected Areas Common.*, vol. 25, no. 2, pp. 340-354, Feb. 2007.

[12] P. Liu, Z.Tao, and S. Panwar, "A Cooperative MAC Protocol for Wireless Local Area Networks",*Proc. IEEE ICC*,June 2005.

[13] J. N. Laneman, G. W. Wornell, and D. N. C. Tse,"An Efficient Protocol for Realizing Cooperative Diversity in Wireless Networks", *Proc. IEEE ISIT*, Washington, DC, June 2001, p. 294

[14] Thanasis Korakis, Zhifeng Tao, Shashi Raj Singh, Pei Liu, Shivendra S. Panwar,"Implementation of Cooperative MAC Protocol : Performance and Challenge in real environment", *EURASIP - Journal on Wireless Communication and Networking*, March 6,2009.

[15] Thanasis Korakis, Sathya Narayanan, Abhijit Bagri, Shivendra Panwar ,"Implementing a Cooperative MAC Protocol for Wireless LANs",*Proc. of IEEE ICC'06*, June 2006.

[16] Key Liu., *"Understanding the implementation of IEEE MAC 802.11 standard in NS-2"*.

[17] K.J.Rayliu, Ahmed K. Sadek, Weifengsu, Andres kwastinski, *"Cooperative Communications and Networking"*,Cambridge University Press.

[18] Ns Manual: New wireless trace file format,
`http://www.isi.edu/nsnam/ns/ns-documentation.html.`

[19] Padmini Misra, *"Routing Protocols For Ad Hoc Mobile Wireless Networks"*.

[20] Andrew S. Tanenbaum. *"Computer Networks"*,4th Edition. PRENTICE HALL, 2002