# DEVELOPMENT OF MULTICAST ALGORITHM FOR THE DETERMINATION OF MINIMUM COST OF BANDWIDTH OVER CODED PACKET WIRELESS NETWORK

**BY**

**SOJA STEPHEN JOSEPH**
**B.ENG (F.U.T.Y) 2006**
**MSc/ENG/36818/2012-2013**

**A DISSERTATION SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES, AHMADU BELLO UNIVERSITY, ZARIA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF MASTER OF SCIENCE DEGREE IN COMPUTER ENGINEERING**

**DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING**

**FACULTY OF ENGINEERING**

**AHMADU BELLO UNIVERSITY,**

**ZARIA, NIGERIA**

**NOVEMBER, 2015**

## DECLARATION

I, **Soja Stephen JOSEPH** declare that the work in this dissertation entitled "**Development of multicast algorithm for the determination of minimum cost of bandwidth over coded packet wireless network**"has been carried out by me in the Department of Electrical and Computer Engineering. The information derived for the literature has been duly acknowledged in the text and a list of references provided. No part of this dissertation was previously presented for any degree or diploma at this or any other institution.

SOJA STEPHEN **JOSEPH** _____ _____

(Student) Signature Date

## CERTIFICATION

Thisdissertation entitled "**DEVELOPMENT OF MULTICAST ALGORITHM FOR THE DETERMINATION OF MINIMUM COST OF BANDWIDTH OVER CODED PACKET WIRELESS NETWORK"**by Soja Stephen **JOSEPH**meets the regulations governing the award of the degree of Master of Science (MSc) in Computer Engineering of the Ahmadu Bello University, Zaria and is approved for its contribution to knowledge and literary presentation.

| | | |
|---|---|---|
| Dr. S. M. Sani | _____ | _____ |
| Chairman, Supervisory Committee | Signature | Date |
| Dr. S. Garba | _____ | _____ |
| Member, Supervisory Committee | Signature | Date |
| Dr. A. M. S. Tekanyi | _____ | _____ |
| Member, Supervisory Committee | Signature | Date |
| Prof. M. B. Mu'azu | _____ | _____ |
| Head of Department | Signature | Date |
| Prof. K. Bala | _____ | _____ |
| Dean, School of Post-graduate Studies | Signature | Date |

## DEDICATION

This research work is dedicated to God Almighty and to my Saviour Jesus Christ

## ACKNOWLEDGEMENT

I thankGod almighty for His mercies, faithfulness, divine protection and love throughout my life and the period of this research work.

My appreciation goes to my late parents, my lovely Mother, brothers and sisters for all their prayers, financials supports and encouragement throughout my studies. Thank you very much and may God bless you all. I also thank my uncles and aunties for their prayers, financial contributions and encouragement throughout my Master programme.

My extreme appreciation goes to my lovely wife and Daughters(Mariam and Wouyema). You have been the best throughout. Your patience, encouragement, care and love were not taken for granted. Thank you very much.

I sincerely wish to express my profound gratitude and appreciation to my supervisor and mentor Dr. S. M. Sani for valuable contributions and support throughout this research work. I appreciate his encouragement and support from the beginning of his lectures on research methodology and to the end of this research. I am very grateful to him for shaping this research to a new direction and providing insight towards the solution and its contribution to knowledge. May the almighty God reward him for the immeasurable service in ensuring that this Department lives up to its standard.

My appreciation also goes to my co-supervisor Dr. S. Garba for his motivation, assistance and valuable contributions throughout this research work. He gave me the impression that i can complete this program. Thank you very much.

I would also like to acknowledge with much appreciation the crucial role of my co-supervisor Dr. A. M. S. Tekanyi for creating time out of his tight scheduled to critically read and correct my work. I thank you sir for the direction you gave me in this research and all the qualitytimesyou create after working hours to shape my work in the right direction. You are my role model and there is much for me to learn about your commitment, generosity and humility. Thank you very much sir.

Special thanks to my Head of Department, Prof. M. B. Mu'azu for his fatherly advice and direction in my research during my mock presentations and in the seminar room. Thank you for all the advice, encouragement and the time you created for going through my work even though you are not part of the supervisory team. Thank you for assisting me with all the materials needed for my research.

My appreciation goes to Professor B. G. Bajoga, Dr. Y. Jibril, Dr. A. D. Usman, Dr. Sikiru, Dr. J. Y. Oricha, Dr. Abu-Bilal, Dr.Jimoh, Dr. Umar for their valuable contributions during seminar and for creating time out of their tight schedule to go through my manuscripts. I am indepted to all the lecturers of the department, Engr Emmanuel Okafor, EngrTaiwo, Engr A. I. Abdullahi, EngrAdamu, Engr Bashir Sadiq, EngrBasiraYahaya, Engr Elvis Obi, EngrOchai, Engr, Zanna, Engr, S. Musa, Engr M. J.Mu'azu, EngrOkorie, Engr Yusuf Ibrahim and all those whose names could not be mentioned. I personally thank Mal. AbdullahiTukur for all the administrative support rendered to me during my master program here in this department.

My profound appreciation goes to my guidance and elder brotherEngr (Dr)MalachySumaila and family. I am shorts of words to express my heart felt gratitude. Thank you for taking me as your younger brother and for accommodating me throughout my master's programme. Thanks a lot for the financial support and for thepieces of advice. May God bless you abundantly.

To EngrGadzama and family, thank you for encouraging me to come and study in ABU Zaria. I truly appreciate the immeasurable concern, prayers, contributions and constant phone calls. Special thanks go to Dr. Samiala Umar of Mechanical Engineering Department. Thank you very much for all the necessary supports and encouragement.

I thank my former General Manager,Mr. Ishaya N.U.Adamu for the encouragement he gave me to further my studies and for his assistance. I equally thank my former Manager Engineering, EngrDanIyaYakubu for his fatherly advice and above all, for allowing me to come here and study. May God bless you all and grant you all your heart desire.

A special thanks goes to the Manager Engineering services, Engr Bello BobbioUba for his understanding and contributions to the success of this programme. I thankthe Chief accountant, Mr. IshayaHannaniyaUba and his assistant Mr. Isaac Wamanta for their words of advice and encouragement. To all my colleagues in Engineering DepartmentEngr Sunday Fabain, Engr Muhammad Musa, EngrRikwinyan Joshua Hassan, HabibaAliyuMaiha, Ibrahim Abdulhamid, Musa Gana, IshakuLivinusKwada, Sinabiji Festus, Mahmud Saleh, Aliyu, Engr Emmanuel Kandi and Sunday Didi for covering me in my place of work whenever I am away for studies.

I am highly indebted to my friends, in-laws and well-wishers.

**Joseph Stephen Soja**

**November, 2015**

## CONTENTS

**TITLE PAGE** i

**DECLARATION** ii

**CERTIFICATION** iii

**DEDICATION** iv

**ACKNOWLEDGEMENT** v

**CONTENTS** ix

**LIST OF FIGURES** xi

**LIST OF TABLES** xiii

**LIST OF ABBREVAITION** xiv

**ABSTRACT** xv

**CHAPTER ONE: INTRODUCTION**

1.1 **BACKGROUND OF THE STUDY** 1

1.1.1 **Multicasting Basics** 5

1.1.2 **Multicast Group** 6

1.1.3 **Multicast Goals** 6

1.2 **STATEMENT OF PROBLEM** 7

1.3**AIM AND OBJECTIVES** 7

1.4 **METHODOLOGY** 8

1.5**SIGNIFICANT OF THE RESEARCH** 9

1.6**DISSERTATION ORGANIZATION** 9

**CHAPTER TWO: LITERATURE REVIEW**

2.1 **INTRODUCTION** 10

2.2 **REVIEW OF FUNDAMENTAL CONCEPTS** 10

2.2.1 **Techniques used for Wireless Multicasting** 10

2.2.1.1 *Network Coding Techniques* 10

2.2.1.2 *Random Linear Network Coding (RLNC)* 14

2.2.1.3 *Opportunistic Network Coding* 16

2.2.2**Optimization** 16

2.2.2.1 *Optimization Approaches* 17

2.2.3**Multicasting in wired and wireless Networks** 19

2.2.4**Multicast Routing** 21

2.2.4.1 *Shortest Path Tree Algorithm* 21

2.2.4.2 *Minimum cost Tree Algorithms* 22

2.2.5 **Multicasting in Coded Packet Networks** 22

2.2.5.1 *The Multicast Network Model* 24

2.2.6**Performance Metrics that Affects Multicast** 27

2.2.7**Shannon-Hartley Channel Capacity Theorem** 29

2.2.8 **Review of the Existing Network Coding Algorithm** 30

2.2.9 **The Improved Network Coding Algorithm** 31

**2.3 REVIEW OF SIMILAR WORKS** 32

**CHAPTER THREE: METHODS AND MATERIALS**

**3.1 INTRODUCTION** 40

**3.2 MATHEMATICAL MODELS FOR MINIMUM COST OF MULTICAST** 40

**3.2.1 Cost Optimization in Coded Packet Wireless Networks** 41

**3.2.2 Mixed Integer Linear Programming Formulation** 42

3.3**COMPARATIVE ANALYSES OF NCA AND THE INCA** 50

3.4 **INSTALLATION AND CONFIGURATION** 51

3.5 **SIMULATION** 51

3.5.1 **Simulation Parameters** 52

**CHAPTER FOUR: RESULTS AND DISCUSSION**

4.1 **INTRODUCTION** 54

4.2**PERFORMANCE EVALUATION OF THE TWO ALGORITHMS**54

4.3 **PERFORMANCE EVALAUTION OF THREE ALGORITHMS** 61

**CHAPTER FIVE: CONCLUSION AND RECOMMENDATION**

5.1 **INTRODUCTION** 69

5.2 **CONCLUSION** 69

5.3 **SIGNIFICANT CONTRIBUTION** 70

5.4 **LIMITATIONS** 71

5.4 **RECOMMENDATIONS FOR FUTURE WORK** 71

**REFERENCES** 72

**APPENDIX A: Code** 80

**APPENDIX B: Formation of A_Z Matrix** 82

**APPENDIX C: The Main Algorithm** 86

**APPENDIX D: The Network Coding Algorithm** 89

**APPENDIX E: Formation of A_Z Matrix (with three parameters)** 99

**APPENDIX F: Test file of the Randomly Generated Nodes by the MIPA** 95

## LIST OF FIGURES

Figure 1.1: Basic Multicasting Concept 2

Figure 2.1: Connections with other disciplines 11

Figure 2.2: Network Coding Based Multicast 13

Figure 2.3: Network Coding Techniques 15

Figure 2.4: The Steiner Tree 19

Figure 2.5: Minimum spanning Tree 20

Figure 2.6: Multicast in Coded Packet Network 23

Figure 2.7: A Network Model 24

Figure 3.1: Flow Chart of Network Coding Algorithm 48

Figure 3.2: Flow Chart of the Developed Algorithm 49

Figure 4.1: Cost of Bandwidth during Multicast for 20 Randomly Generated Nodes 55

Figure 4.2: Cost of Bandwidth during Multicast for 30 Randomly Generated Nodes 56

Figure 4.3: Cost of Bandwidth Multicasting 40 Randomly Generated Nodes 57

Figure 4.4: Cost of Bandwidth Multicasting 50 Randomly Generated Nodes 58

Figure 4.5: Cost of Bandwidth Multicasting 60 Randomly Generated Nodes to groups of Receivers. 59

Figure 4.6: Cost of Multicasting 70 Randomly Generated Nodes to Groups of Receivers 60

Figure 4.7: Cost of Bandwidth during Multicast for 20 Randomly Generated Nodes 62

Figure 4.8: Cost of Bandwidth during Multicast for 30 Randomly Generated Nodes 63

Figure 4.9: Cost of Bandwidth during Multicast for 40 Randomly Generated Nodes 64

Figure 4.10: Cost of Bandwidth during Multicast for 50 Randomly Generated Nodes 65

Figure 4.11: Cost of Bandwidth during Multicast for 60 Randomly Generated Nodes 66

Figure 4.12: Cost of Bandwidth during Multicast for 70 Randomly Generated Nodes 67

## LIST OF TABLES

Table 4.1: Costsof Bandwidth for 20 Randomly Generated Nodes Multicasting to Groups of Receiver 55

Table 4.2: Costsof Bandwidth for 30 Randomly Generated Nodes Multicasting to Groups of Receivers 56

Table 4.3: Costsof Bandwidth for 40 Randomly Generated Nodes Multicasting to Groups of Receivers 57

Table 4.4: Costsof Bandwidth for 50 Randomly Generated Nodes Multicasting to Groups of Receivers 58

Table 4.5: Costsof Bandwidth for 60 Randomly Generated Nodes Multicasting to Groups of Receivers 59

Table 4.6: Costsof Bandwidth for 70 Randomly Generated Nodes Multicasting to Groups Receivers 60

Table 4.7: Costsof Bandwidth for 20 Randomly Generated Nodes multicasting to Groups of Receivers 61

Table 4.8: Costs of Bandwidth for 30 randomly Generated nodes multicasting to Groups of Receivers 62

Table 4.9: Costs of Bandwidth for 40 Randomly Generated Nodes multicasting to Groups of Receivers 63

Table 4.10: Costs of Bandwidth for 50 randomly generated nodes multicasting to Groups of Receivers 64

Table 4.11: Costs of Bandwidth for 60 randomly generated nodes multicasting to Groups of Receivers 65

Table 4.12: Costs of Bandwidth for 70 randomly generated nodes multicasting to Groups of Receivers 67

## LIST OF ABBREVIATIONS

| Acronyms | Definition |
|---|---|
| NCA | Network Coding Algorithm |
| INCA | Improved Network Coding Algorithm |
| MIPA | Multicast Incremental Power Algorithm |
| MIP | Mixed Integer Linear Programming |
| IPTV | Internet Protocol Television |
| RLNC | Random Linear Network Coding |
| ONC | Opportunistic Network Coding |
| LP | Linear Programming |
| IP | Integer Programming |
| VoIP | Voice over Internet Protocol |

## Abstract

Minimum cost of multicast is the process of finding the node transmission cost such that the total cost of bandwidth used for sending messages from source to any destination nodes is minimized. The Network Coding Algorithm(NCA) was developed using linear programming framework by considering packets delay as the only key performance metrics. This research work is based on Shannon Hartley channel capacity theorem aimed at interpreting the cost of bandwidth within the channel and this was achieved by considering more performance metrics that affect the channel. Loss and delay of packetare collectively considered and later delay, loss and rejection packet are also considered on the Network Coding Algorithm. The problem was formulated as optimization problem using mixed integer linear programming problem. Simulation results showed that the average cost of bandwidth decrease as more performance metrics are collectively considered. When two performance metrics of delay and loss are considered collectively, there was a reduction in the average cost of bandwidth used during multicast by 27%, 28.7%, 29.5%, 20.9% 18.5%, 23.8%, 21.6% and 20.9% for 20, 30, 40, 50, 60 and 70 randomly generated nodes multicasting to groups of receivers when compared with the benchmark algorithm of delay only. Similarly, considering three performance metrics (delay, loss and rejection) achieved a reduction in the average cost of bandwidth by 37.07%, 37.6%, 41.15, 29%, 30% and 35.3% for 20, 30, 40, 50, 60 and 70 randomly generated nodes multicasting to various groups of receivers as compared with benchmark algorithm. Equally, results also showed that the average cost of bandwidth for NCA with three performances parameters decreases for 12.7%, 13.6%, 16.6%, 11.1%, 16.3% and 15.1% for all the number of randomly generated nodes as compared to a situation when two performance metrics are considered. From the results obtained, it can be observed that the cost of bandwidth with 40 randomly generated nodes recorded the highest decrease in the average cost of bandwidth because the cost of multicast for NCA, INCA with two and three performance metrics tends to be linear and increase as the number of receivers increases.From these results, it could be concluded that the more performance metrics considered in the NCA, the better the true picture of the channel is represented. When interpreting the cost of bandwidth using Shannon-Hartley channel capacity theorem, the study established that the cost of bandwidth within the channel was under estimated as more performance metrics are considered.

# CHAPTER ONE
# INTRODUCTION

## 1.1 BACKGROUND

The process of routing messages from a source to a set of destination nodes is known as multicast (Wen & Liao, 2010). According to Xuan and Lea (2011), Multicast video streaming and video conferencing is speedily becoming a key requirement of many computer networks supporting numerous multimedia applications. In multicast, a single message is delivered to a cluster of destination in a network and this concepts has been studied for both wired and wireless networks (Lertpratchya *et al.*, 2014). Multicast as a means of communication is an essential part of many next generation networks and there are limited numbers of network usersthat support multicast in the Internet today. This has made it necessary for applications requiring multicast services to obtain services at a higher standard. Also, according to Chopra and Mishra (2012), multicasting is the transmission of data in a group of nodes which is recognized by one and distinctive address. Multicast services allow one source to send information to a large number of receivers and it finds application in many areas such as audio conferencing, video conferencing, video-on-demand services, peer-to-peer file sharing, electronic learning, online shopping, distributed interactive simulation, software enhancement, Wireless Internet Protocol Television (IPTV), distance Education, group-oriented mobile commerce, firmware reprogramming of wireless devices, distributed database replication etc, (Han *et al.*, 2014; Haripriya & Valluvari, 2013; Wu *et al.*, 2014). Ashlwede *et al.*, (2000) showed that multicasting can take place in a situation where two source nodes are simultaneously sending information to different groups of receivers by superposition. Figure 1.1 shows the basic concept of multicasting.

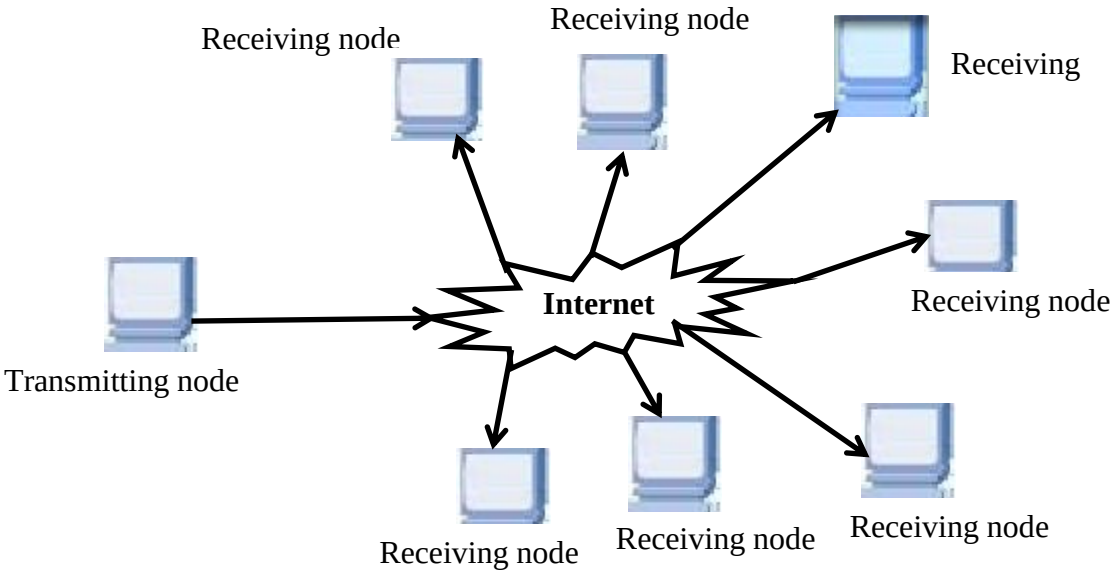



Figure 1.1: Basic Multicasting Concept (Karaoglu & Heinzelman, 2010)

Multicast communication is a process of passing information over large networks and it saves a lot of time and resource (Annapurna *et al.*, 2014). Wireless networks has multicast advantage that is, if data is transmitted from node i to node *j*, all the nodes whose distance *i* is lesser than *j* receive the data for free before reaching to the targeted recipient. One of the major challenges of wireless multicast is to be able to support a large number of users while simultaneously taking care of packets delay, packets loss, packets congestion and low feedback overhead (Wu *et al.*, 2014). One of the principal resources that an overlay network must manage is the access bandwidth to the Internet and the energy consumption during multicast as nodes at different locations communicate with each other by relaying information over wireless links(Mani, 2014). This represents a major cost, and is typically the resource that constrains the number of simultaneous multicast sessions that an overlay network can effectively support(Xuan & Lea, 2011).

Multicasting over wireless networks is a significant but at the same time is a challenging goal in the field of wireless communication networks(Lun *et al.*, 2006).

Itrequires many issues to be addressed such as bandwidth, topology, loss of packets, delay, routing, reliability, security and quality of service, before it can be fully deployed (Qadir *et al.*, 2014). Providing a trade-off between stability, throughput, delay, and packet loss with reduced bandwidth requirement and less power consumption is the aim of multicasting in both wire line and wireless networks. Multicasting is a more general form of group communication than broadcast because all broadcast session can be viewed as a special case of multicast where all nodes are within the multicast group(Zongpeng, 2008). In several scenarios, there is a great challenge of cost allocation with broadcasting most especially when the users are linked to a common source at a minimum cost. Network coding helps in increasing network throughput and security by encoding several packets with single packet size length and forwarding them in a single transmission time slot (Jain & Mahdian, 2007).

In order to avoid sending several copies of the same message through the same link,multicast routing is used and this can be achieved by using a tree connecting the receivers to the same source. Multicasting in a larger network is always linked with many challenges such as bandwidth and energy consumption, and this is as a result of multiple users who probably have conflicting purposes (Rajkumar, 2014; Ramamoorthy, 2011). It is necessary to note that the expected cost of using any dynamic network might change due to the instability of the multicast members and other factors such as distance, location and differences in continents(Skorin-Kapov, 2001). The problem of finding node transmission cost such that the total cost is minimized as an optimization problem is referred to as minimum cost multicast (Wen & Liao, 2010). Different optimizations goals have been formulated and applied in multicast algorithms to minimize bandwidth and energy consumption, and also to

determine what constitutes a good multicast tree. One of such goal is providing minimum delay along the tree, which is an important factor for some multimedia applications such as real time audio and video conferencing. Minimizing the total bandwidth utilization of the links and energy consumption is the goal for all optimization problems for multicast routing algorithms(El Rouayheb *et al.*, 2006).

In network-wide broadcasting the objective is to distribute the generated data to all the nodes in the network. This problem is similar to the scalability problem that arises when trying to broadcast large amounts of data to many end-users in real time, for example, live streaming on the internet(Lagercrantz, 2014). The main difference is that multicast, as opposed to broadcast, is directed to a specific set of end-users. Also, broadcasting everything to everybody would violate security principles of sensitive data (Lagercrantz, 2014). It would also generate unnecessary data traffic which increases the operating costs. On the other hand, the objective of multicasting is to deliver the data to a subset of the nodes in the network.

There is a multitude of ways to achieve the minimum cost, which depends mostly on the characteristics of the network being used. If routers are only able to forward packets , the weighted shortest paths to each sink are the most cost efficient way of delivering the packets (Ajibesin *et al.*, 2013). Ad-hoc and coded packet networks supports many multicast applications such as distributed data processing, Internet telephone, interactive multimedia conferencing, and real time video broadcasting (Ramamoorthy, 2012; Rashed, 2012). A packet network is a network in which raw information bits are exchanged and most often, voice quality is affected by delay, delay jitter, packet loss and unreliable packet delivery all of which are unusual characteristics of the essential network service.

These characteristics lead to variations in packet delivery, which are the most significant concerns of voice over internet (Karthikeyan & Shankar, 2014). Voice over internet protocol (VoIP) is expected to impact on some of the better developments in higher education, for instance, virtual universities, personality management and so on. The fundamental process carried out in a VoIP call are, transformation of the caller's analog voice signal into a digital format, compression and transformation of the digital signal into discrete Internet Protocol packets, Transmission of the packets through the Internet or other IP-based network and Reverse transformation of packets into an analog voice signal for the call recipient (Karthikeyan & Shankar, 2014). A multicast group has only one source node and multiple destination nodes; each pair of nodes is connected by a logical edge with the minimum cost between the nodes (Hu & Zhang, 2013). Two models are usually associated with multicast; the node-based model and link based model. The node based model is mostly used during multicasting because single transmission can be heard by several nodes rather than using a link based model where a node transmits separately to each of its neighbors resulting in increased cost of operation (Xiong & Li, 2010).

### 1.1.1 Multicasting Basics

The basic modules of a multicast routing include sources, receivers, and a group having direct or indirect connections with each other and the opinion of the group is essential to the concept of multicasting. The originator of the data stream is called a 'source' while the end-user that wishes to receive the data stream is called the 'receiver'. In multicast routing, the source is interested in transmitting its data stream to all the receivers' within the multicast group (Qadir *et al.*, 2014). For point-to-

multipoint group communication, multicasting comes across as the most efficient means of communication(Lagercrantz, 2014).

### 1.1.2 Multicast Group

A multicast group is a set of nodes in the networks that share the same piece of information. It can have one or more source nodes, and more than one destination. A multicast group can be static or dynamic. Static group cannot be changed after its creation and it is frequently modeled as a Steiner tree problem. On the other hand, dynamic groups can have members added and removed at any time (Oliveira & Pardalos, 2005).

### 1.1.3 Multicast Goals

In wireless networks, multicasting can have various performance goals depending on what the user want. In particular it is important to provide high quality of service, cost efficient performance, and to ensure reliability and security (striegel & Manimaran, 2002). In many applications such as teleconferencing, distance learning, and online shopping exploit multicast transmissions while requiring a certain minimum quality of service for satisfactory performance (Paschos *et al.*, 2014). In other to support such applications, a network support for scalable group-based applications with strict quality of service requirement becomes paramount. Various performance metrics (such as packet delay, delay jitter, packet loss, throughput, packet congestion e.t.c) can be used to quantify the quality of service (Wenzhong & Zhenyu, 2013). Ensuring reliability and security in multicast routing is an important goal for multicast protocols(Lagercrantz, 2014).

## 1.2 STATEMENT OF PROBLEM

Bandwidth utilization during multicast over coded packet wireless networks has become an area of intensive research in trying to establish the true cost of bandwidth in order to achieve the true picture of the channel. The problem that prompted this intensive research ever since is to be able to determine those factors which seriously affect the cost of bandwidth during multicast over coded packet wireless networks. Another crucial issue investigated is whether the channel capacity is truly represented in the proposed algorithm in accordance with Shannon Hartley channel capacity theorem.

Therefore, there is a need to find an improvement to the proposed algorithm, the approach taken by this research by introducing more performance parameters which affect the channel and particularly bandwidth utilization. To do this, two parameters NCA is used where delay and loss of packets are considered and then that with three metrics of delay, loss and rejection of packets. These are then compared to the NCA of one parameter of delay only to be able to draw a conclusion.

## 1.3 AIM AND OBJECTIVES

The aim of this research is the development of an improved multicast Algorithm for interpreting the cost of bandwidth within the channel using Shannon-Hartley channel capacity theoremby considering more performance metricsthat affect the channel.

The objectives of the research are as follows:

i. To examine the effect oftwo and threeperformance metrics of Network Coding Algorithm onmulticasting over coded packet wireless networks.
ii. To develop an INCA by considering more performance metrics that affects the cost of bandwidth within the channel using linear programming framework.

iii. To establish the true picture of the channel in terms of the cost of bandwidth in accordance with Shannon Hartley channel capacity theorem when a lesser number of performance parameters are considered in the NCA.

## 1.4 METHODOLOGY

The following methodology is adopted in carrying out this research and is itemized as follows.

i. Determination of performance metrics (packets delay, packets loss, packets congestion, packet rejection and throughput) that critically affects the cost of bandwidth during multicast over coded packet wireless network in order to know the true picture of the channel.

ii. Formulation of the problem as an optimization problem using mixed integer linear programming with suitable constraints so as to improve the performance of the Network Coding based-Algorithm for effective bandwidth utilization during multicast.

iii. Using only two and three performance metrics to improve the performance of existing network coding based-algorithm in order to determine their impact within the channel.

iv. Comparison of the performance of the existing Network Coding Algorithm with the improved Network Coding based-Algorithm using Visual studio 2010 using the same simulation patterns.

v. Validation of the performance of the improved network coding based-algorithm with the existing network Coding algorithm using Matlab 2010b.

## 1.5 SIGNIFICANT OF THE RESEARCH

This research is significant in the field of Engineering because over the years, multicasting over wireless network is a great challenge due to the high rate of bandwidth consumption. When the research parameters (loss, delay and rejection of packets) are implemented and controlled in real time, the work will go a long way to minimize the consumption of bandwidth and it will also encourage group communication over the internet. This is what this research is set to achieve.

## 1.6 DISSERTATION ORGANIZATION

Chapter one presents the general introduction. The rest of the chapters are as follows: detailed review of similar works and relevant fundamental concepts pertinent to Minimizing cost of multicast, multicast routing and techniques are presentedin Chapter Two. Mathematical models, formation of the problem as mixed integer linear programming describing the performance metrics considered are presented in Chapter Three. Chapter Four presents results, their analysis and discussions. Conclusion and recommendation are presented in Chapter Five and finally, all the references quoted in this research work and appendices are also provided at the end of the thesis.

# CHAPTER TWO

# LITERATURE REVIEW

## 2.1 INTRODUCTION

This section reviews pertinent theoretical concepts and similar research works and it is divided into two parts:

(i) review of fundamental concepts and,

(ii) review of similar works.

Each of these elements will be discussed in section 2.2 and 2.3 respectively in this main section two.

## 2.2 REVIEW OF FUNDAMENTAL CONCEPTS

This subsection presents the review of the fundamental concepts pertinent to this research work**.** These concepts includemulticasting in coded packet networks, network coding techniques and the algorithms used during multicasting.

### 2.2.1Techniques used for Wireless Multicasting

Various techniques from diverse fields have been used in the design and analysis of wireless multicast. Some key ones that are related to this research work are network coding techniques and optimization.

#### 2.2.1.1 *Network Coding Techniques*

Network coding promises to offer benefits along very diverse dimensions of communication networks, such as throughput, wireless resources, security, complexity, and resilience to link failures(Ahlswede *et al.*, 2000). Networked systems arise in various communication contexts

such as phone networks, the public Internet, peer-to-peer networks, ad-hoc wireless networks, and sensor networks.

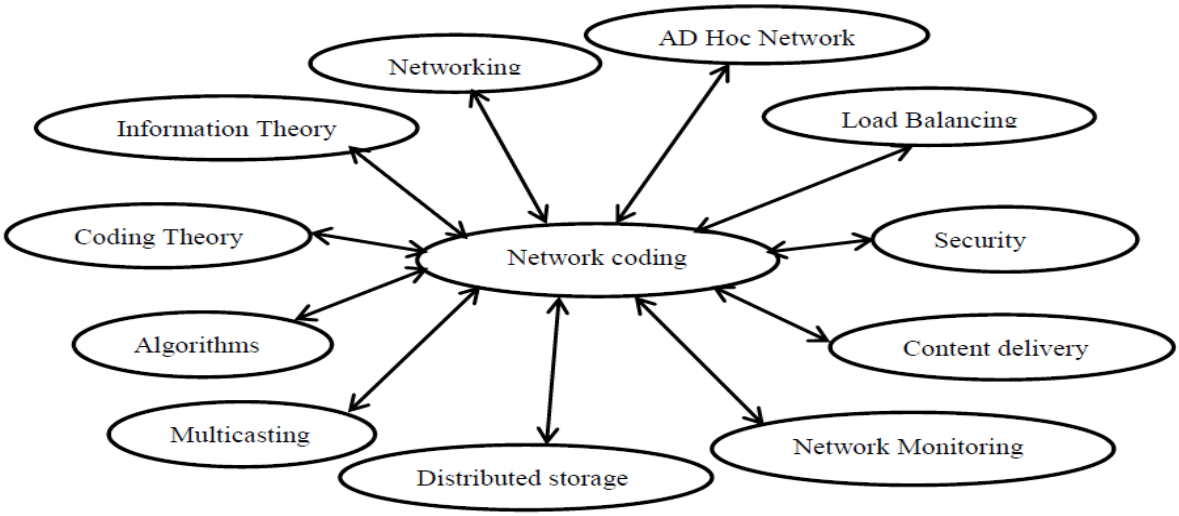



Figure 2.1: Connections with other Disciplines ((Fragouli & Soljanin, 2007)

Ahlswede *et al.,* (2000) addressed multicast problem in both single source and multiple source and showed how network coding has been employed to achieve optimality. Also, coding by superposition can take place when multiple source multicasting is used to achieve optimality when the information sources are independent and are coded separately.The interest in Network coding increase as new applications of ideas are discovered in both the theory and practice of networks, and realize new connections with many diverse areas (Fragouli & Soljanin, 2007; Sanders *et al.*, 2003). Network coding outperformed the traditional technique for multicasting in a computer network by distributing the information through the network from the source to the destination nodes (Ahlswede *et al.*, 2000). Multicast with coding takes place in directed networks which is a situation where the nodes are connected to edges and the edges are in direction of the nodes.Sending linear combinations of packets offers a good advantage of multipath diversity for security against wiretapping attacks (Fragouli & Soljanin, 2007). There is a very rich history in using Linear Programming (LP) for studying flows through networks.Network coding has also been studied

within this framework, which is the most efficient method for addressing networks with costs. Network coding utilizes cheap computational power to dramatically increase network throughput. Instead of forwarding the original packets directly, network coding allows the source node to linearly combine multiple packets together (Tan *et al.*, 2012). Alternatively, network coding has stretched out the functionality of network nodes from forwarding packets to accomplishing algebraic operations on received data (Zhao *et al.*, 2014). With this additional function, each transmitted packet has almost the same contribution in reconstructing the original packet together (Norde *et al.*, 2004).

Network coding entails the transmission, encoding and re-encoding of messages arriving at nodes inside the network, such that the transmitted messages can be decoded at their final destinations(Bassoli *et al.*, 2013). It finds application in real networks where packets are transmitted in form of bit and are subjected to random loss and delay of packets. The edges have variable capacities due to congestion and other cross traffic, node and link failures (Anshelevich *et al.*, 2008; Debbra, 2003). It is also relevant in wireless and ad hoc multichip networks, sensor networks, stationary wireless residential or community mesh networks, and so on.

Network coding approach has presented a better way of processing information, which is referred to as store, process and forward (Dimakis *et al.*, 2010). The technique allows the intermediate node to generate new packets from the nodes and to a large extendit improves the network throughput and reliability (Han *et al.*, 2014).It also improves network security because adversary node can combine packets using secure coefficient to enhance privacy and increase data integrity (Hamza *et al.*, 2013).

Network coding is one of the techniques that explore the challenges of multicast networks and is considered for minimum cost of multicast over a coded packets

network (Ajibesin *et al.*, 2013; Yin *et al.*, 2014). Network coding was first presented by Ahlswede *et al.*, (2000) as a practical example showing information flows during multicast. It saves bandwidth and increases the network's throughput. As shown in Figure 2.2; the source node '*s*'sends packets $p_1$ and $p_2$ to receivers' $r_1$ and $r_2$, respectively. Both routers *a* and *b* broadcast the received packets $p_1$ and $p_2$ to receiver $r_1$ and $r_2$, respectively. A network coding based router c therefore performs the binary sum of the two packets and output the packet $p_1 \otimes p_2$to router 'd' with broadcast capability .

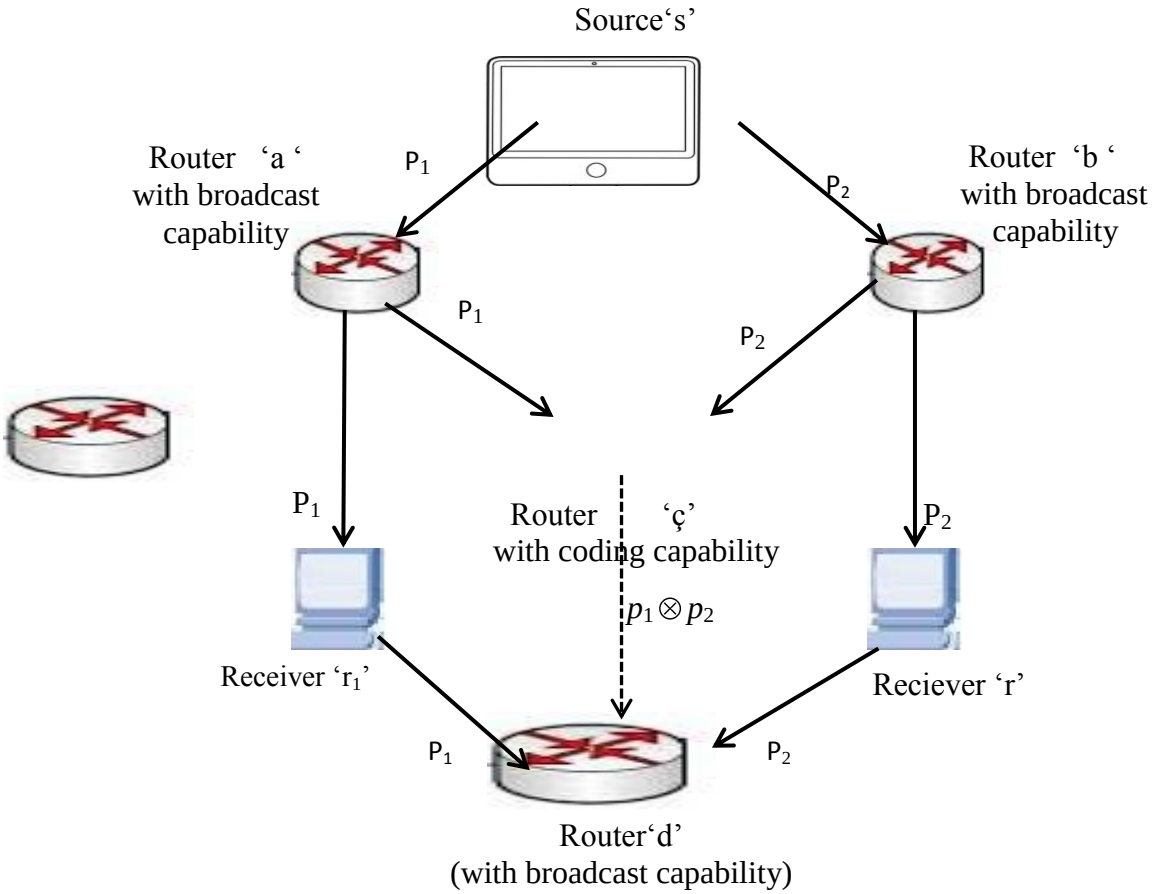


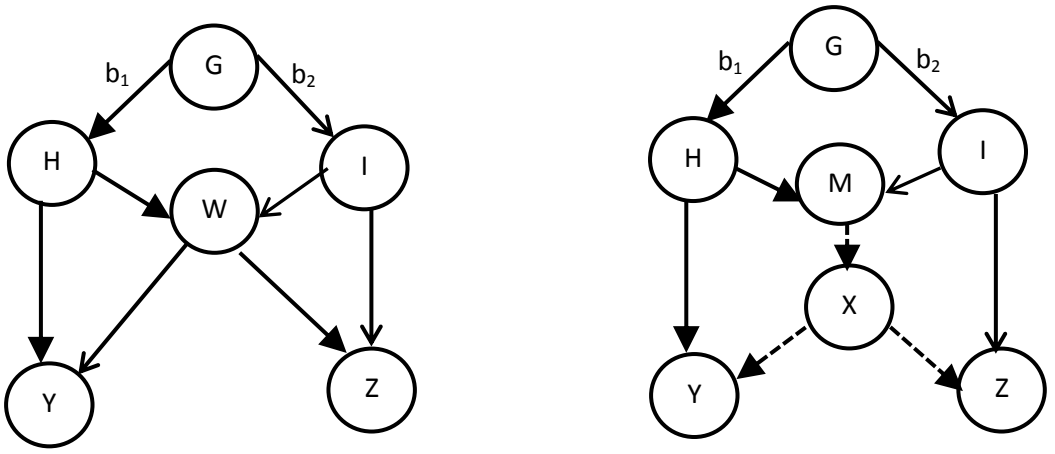

Figure 2.2: Network Coding Based Multicast (Ajibesin, *et al.*, 2013)

In the network model of Figure 2, router *c* is designed with network coding capability and this operation is known as *XOR*ing, which is one of the simplest forms of data coding.

Further, the receiver $r_1$ receives packet $p_1$ and coded packet $p_1 \otimes p_2$ uses the coding scheme to decode packet $p_2$. Similarly, the receiver $r_2$ uses the coding scheme to recover packets $p_1$ from received packet $p_2$ and coded packet $p_1 \otimes p_2$. It can be concluded that without the coding scheme, it will be definitely impossible to multicast two packets per unit time from the source $s$ to both the receivers' $r_1$ and $r_2$ (El Rouayheb, *et al.,* 2006). Two important classes of network coding for multimedia application distinguished in the literatures are linear network coding and opportunistic network coding.

2.2.1.2*Random Linear Network Coding (RLNC)*

Network coding is an emerging solution that can significantly improve the throughput and power efficiency of wireless networks by allowing mixing of various traffic flows via algebraic operations. With network coding, however, a packet has to wait to be network-coded with others given the stochastic nature of the packet arrival process of the various flows. This may result in large delay and packet-loss rate. Random Linear network coding is interested in how fast eachnode can receive the complete information, or equivalently, the information rate arriving at eachnode (S. Y. Li *et al.*, 2003). This is implemented by combining packet with independent, random and non-zero coefficient. In RLNC, finite amount of information is generated at the source and multicast to other nodes on the network and every node can pass on any of its received data to other nodes. At each non-source node which serves as a sink, the complete information generated at the source is recovered. Linear network coding is aimed at how fast each sink node can receive the complete information and how a node can assist its neighbor by sending a network coded packet including packets of its own and its neighbor (L. Ma *et al.*, 2013). As an example, consider a network multicasting two data bits, $b_1$ and $b_2$, from the source as shown in Figure 2.3




(a) Network without LNC

(b)Network with LNC

Figure 2.3: Network Coding Techniques (Ma, *et al.,* 2013)

With linear network coding, it is possible to multicast two bits $b_1$ and $b_2$ from one source S to destination nodes Y and Z. In figure 3(a) channels GH, HY, HW and WZ carry bit $b_1$ and channels GI, IZ, IW, and WY carry the bit $b_2$. In this scheme, an intermediate node sends out a data bit only if it receives the same bit from another node, For example, the node H receives the bit $b_1$ and sends a copy on each of the two channels HY and HW. Also, the node I receive the bit b2and send a copy into each of the two channels IW and IZ. Linear network takes place in Figures 2.3(b) and it is often used in coded packet networks. Channels WX, XY, XZ carry the exclusive-OR $b_1 \otimes b_2$ and the node Y receives $b_1$ and $b_1 \otimes b_2$, from which$b_2$ can be decoded. Also, the node Z can decode the bit $b_1$from $b_2$ and $b_1 \otimes b_2$ (S. Y. Li *et al.,* 2003).RLNC performs coding across packets of the same sessions and it attracts more attention as it independently and randomly selects linear mapping from inputs onto output links. This encoding scheme has been implemented in practice, which is widely applied to network security, and network monitoring, multicasting and management (Zhao *et al.,* 2014).

2.2.1.3*Opportunistic Network Coding*(ONC)

In ONC,packets are coded at different sessions and each node uses knowledge of what its neighbors have overheard such that each encoded packet can be decoded immediately at the next stage (Cui *et al.*, 2008). Packet combinations are selected according to the received and lost packet state of each link where nodes often opportunistically overhear packets that are not designated to them (Chen *et al.*, 2007). This overheard packets become the side information for the nodes (Dau *et al.*, 2013). It is the efficient solution of packet delivery for wireless networks (Dougherty *et al.*, 2014).

### 2.2.2Optimization

Optimization techniques are used to find a set of design parameters that give the best possible result. Optimization theory comprises of tools and techniques for determining optimal decision in scenarios which may also incorporate certain constrains. It deals with the problem of minimizing or maximizing a certain function which is usually determined by functional inequalities. There are many classes of optimization problems generally characterized on the basic of their objective functions and constraint (Qadir *et al.*, 2014). It can be linear, nonlinear or convex optimization. Nonlinear optimization Toolbox provides widely used optimization algorithms for solving nonlinear programming problems in MATLAB(Rardin, 1998). An important characteristic of an optimization problem is its discrete or continuous nature.

Bandwidth optimization can be achieved by using different programming frameworks. Each of programming techniques has different characteristics and can address different problems. Hence, the need for their analysis to determine the most suitable for use in this work(Bley, 2011). At times a combination or integration of

these programming techniques could provide better solution to the research problem. This is the case in this study where linear programming framework formulated as a mixed integer programming problem.

2.2.2.1 *Optimization Approaches*

i. **Convex program:** Convex optimizationmarks the biggest progress in the field of optimization since the formulation of linear programming and invention of the simplex algorithm (Nemirovski, 2006). Convex optimization subsumes least square, linear optimization, conic programming and geometric programming classes of optimization (Boyd & Vandenberghe, 2004; Luo & Yu, 2006).Formulating a problem as a convex optimization problem can be rewarding for two important reasons(Bhadra *et al.*, 2006). Firstly, a convex optimization problem can be reliably and efficiently solved through special methods such as the interior point method proposed for linear optimization (Qadir *et al.*, 2014). And secondly, duality theory which often reveals an interesting interpretation of the primal problem and often leads to an efficient distributed method for solving it.

ii. **Linear programming (LP):** Linear programming is a problem of combinatorial optimization and it welcomes polynomial-time solutions. In a linear program, there are *v*ariables, constraints, and an objective function.The variables or decisions take on numerical values. Constraints areused to limit the values to a feasible region. Linear programming is used in finance, energy, operations research, and other applications where relationships between variables can be expressed linearly. Optimization Toolbox includes three algorithms used to solve linear programming problems. They are: the simplex algorithm which is a systematic procedure for generating and testing candidate vertex solutions to a

linear program. These constraints must be linearin the decision variables. The objective function then defines which particularassignment of feasible values to the variables is optimal and it is the one thatmaximizes (or minimizes, depending on the type of the objective) the objectivefunction (Bosch & Trick, 2004).In many optimization problem, it only make meaning when a certain optimization variable takes on integer value, for example, number of packets, distances, flows, etc. (Qadir *et al.*, 2014)

iii. **Integer programming (IP):**Integer programming is used when all the optimization variables must take only integer values. Integer programming problems can belong to a class of linear or nonlinear. If it belongs to the linear class, its objective and constraint functions are both linear, the class of model is referred to as integer linear programming and vice versa for non -linear class programming problem.

iv. **Mixed integer programming (MIP):**Mixed-integer linear programming expands the linear programming problem with the additional constraint that some or all of the variables in the optimal solution must be integers. Mixed Integer Linear Programming theory and practice has been significantly developed and is now an indispensable tool in business and engineering. The reasons for the success of MIP are Linear Programming (LP) based solvers and the modeling flexibility of MIP (Vielma, 2014). Modeling non-convex functions has been a central topic of MIP formulations since its early developments and these problems are often formulated during multicast (Bley, 2011). If in an optimization model, certain optimization variables can only take on integer value while other cannot, this class of optimization model is referred to as mixed integer programming model(Ajibesin *et al.*, 2014).

### 2.2.3 Multicasting in wired and wireless Networks

Multicasting has been a method of supporting group communication which allows transmission and routing of packets to multiple destinations using fewer resources. It can also be described as the processing of routing information from the source node to a set of receivers; $r_1$, $r_2 \ldots r_n$ nodes (Immorlica *et al.*, 2008). The performance of multicasting involves the generation of a multicast tree, which is a typical tree topology that is used for multicast routing and is a tree spanning from the source and destination nodes (Hougardy *et al.*, 2014). To satisfy all the destination nodes, the source node needs to keep sending until the destination node with the worst packet reception link successfully receives all the packets, from the source node (Xiumin *et al.*, 2012). The Steiner tree is a network connecting a set of required vertices with the minimum cost (Zhu *et al.*, 1995).

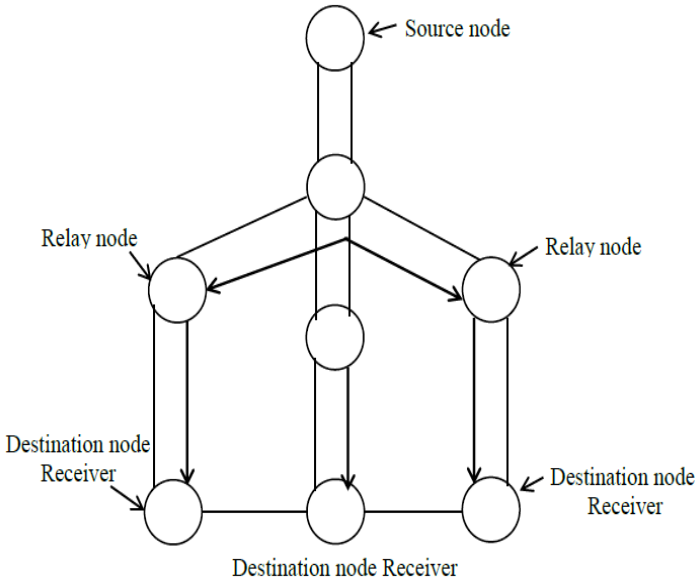



Figure 2.4: The Steiner Tree (Herzog *et at.*, 1997)

Given an undirected graph, G (V, E) with a cost or weight $c_e$ on each edge $e \in E$ and given a subset of the nodes R, called required nodes, the Steiner tree consists of finding a minimum cost sub-tree of G that spans all the nodes in R, this is shown in

Figure 2.4(Herzog *et al.*, 1997).In general Steiner tree problem is an NP-complete problem. NP-complete problems are classes of multicast problems that are impossible to solve effectively using a specific approximation method or algorithm.The cost of a Steiner tree T is indicated by c(T) and it is defined as (Castillo *et al.*, 2002).

$$c(T) = \sum_{e \in T} C_e \qquad (2.1)$$

where; c (T) is the cost of the multicast tree and $C_e$ are the cost of the edges.

A minimum Spanning Tree is a graph with non-negative cost (weight) associated with each edge $e \in E$. It consists of finding a tree with the minimum or lowest total cost that spans to all the nodes of the graph. This is shown in Figure 4.

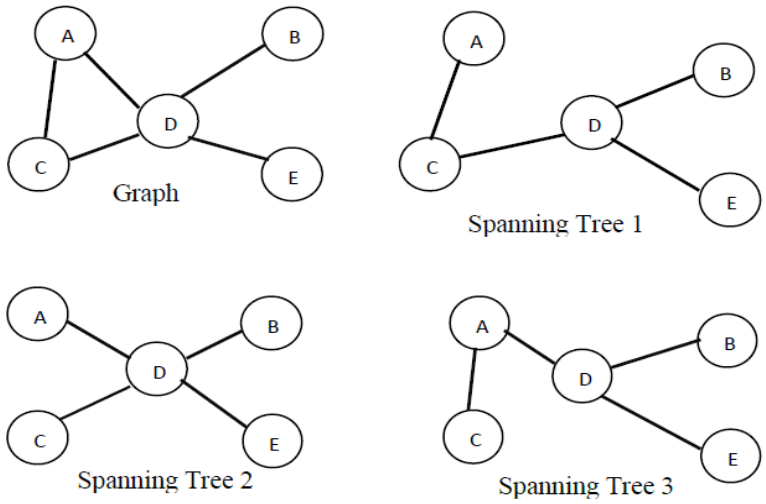



Figure 2.5: Minimum Spanning Tree (Hu and Zhang, 2013)

Prim's algorithm is used to find the minimum spanning tree in graph G (V, E). As shown in Figure 2.5, the graph G spans into three different sub-graphs and if a certain value is assigned to each edge, starting with a node $s \in V$ and a set L:=(s), at each iteration, the algorithm chooses a minimum cost edge $e$ of $u, v \in E$, connecting

node $v \in L$ and updates the set L:=L$\cup$ $v$ . This is repeated until L=V and the tree with minimum value is the minimum cost of the graph.

### 2.2.4Multicast Routing

Based on the literature consulted, there are two fundamental approaches to multicast routing (Zhao *et al.*, 2014): the shortest path tree and the minimum spanning cost tree and various heuristics algorithms have been adopted for both the shortest path tree and the minimum spanning tree for determining cost. The shortest path tree algorithms minimize the distance (or cost) from the sender to each receiver, whereas the minimum cost tree algorithms minimize the overall cost of the multicast tree. The shortest path tree has been widely used for many real time applications because it minimizes the distance from the source node to the destination node.

#### 2.2.4.1 *Shortest Path Tree Algorithm*

The goal of shortest path tree algorithms is to construct a tree rooted at the sender and spanning all the receivers such that the distance between the sender and each receiver along the tree is minimize (Zhao *et al.*, 2014). It is a graph search algorithm that solves the single-source shortest path problem for a graph with non-negative edge path costs, producing a shortest path tree. For a connected, undirected network with one special node, called the source, associated with each edge is a distance, a nonnegative number. The objective is to find the set of edges connecting all nodes such that the sum of the edge lengths from the source to each node is minimized. This is called shortest-paths tree(SPT) rooted at the source (Bang & Chao, 2009). This research considers shortest path tree because multicasting finds application in many multimedia applications.

The goal of the shortest path tree (SPT) algorithm is usually to compute a tree rooted at the sender and spanning to all the receivers such that the distance between the sender and each receiver along the tree is minimize. Bellman-Ford and Dijkstra's algorithm are algorithms known for computing shortest path tree (Hougardy *et al.*, 2014). Dijkstra's algorithm have been used to generate the shortest paths from the source to destination nodes andits provide optimal solution of the multicast tree(Banchs *et al.*, 2014).

#### 2.2.4.2 *Minimum Cost Tree Algorithms*

Minimum cost tree algorithms is focused at minimizing the overall cost of the tree and they are based on the minimum Steiner tree problems, which is NP-complete (Zhao *et al.*, 2014). NP-complete problems are classes of problems that are widely believed that it is impossible to solve effectively using a specific algorithm. By assigning a cost to each edge of the graph, minimum cost tree algorithm compute the tree which minimize the sum of the cost of its edges(Z. Li & Chu, 2012). The cost of an edge (link) can be the packet transmission delay on that link, or the distance between the two vertices(Soumen & Mohammad, 2010).

This problem is known as the minimum spanning tree problem and is an undirected connected tree with weighted graph that is spanning with minimum distance. Several heuristic algorithms such as Prim's algorithm, generic algorithm, kruskal's algorithm, clustering algorithm amongst other algorithms are known for computing minimum spanning tree problem.

### 2.2.5 Multicasting in Coded Packet Networks

In any network model just like in real world networks, nodes are able to forward, copy and direct copies to multiple nodes and there is no reason to limit nodes to only

that functionality. In coded packet networks, nodes are additionally able to use other functions on multiple incoming packets (Schalkamp, 2012). This can be used to combine two packets, for example by using bitwise XOR (A⊗B) in order to reduce the total amount of packets needed to be transmitted through the network (Lopes *et al.*, 2010). This is illustrated in Figure 6. Node Q is the source node that sends packets A and B to node 1and 2 respectively.

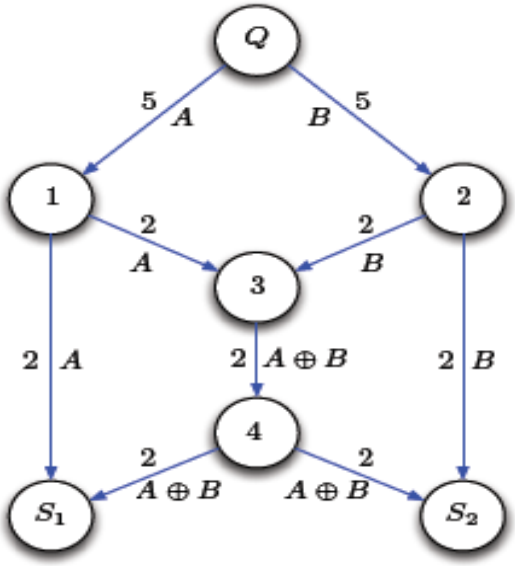



Figure 2.6: Multicast in Coded Packet Network (Lopes, *et. al*., 2010)

Node 3 combines the incoming Packets A and B to the new packet A⊗B and sends it via 4 to both sinks nodes $s_1$ and $s_2$. At the moment, the packet A⊗B is useless for each sink, as they are not able to derive both original packets A and B from it. Therefore sink $S_1$ receives packet A and sink $S_1$ receives packet B, as this enables both sinks to derive the other packet by using XOR on A⊗B and A, and respectively B. This process continues until packets are concurrently transmitted from a source to a set of receivers in a network coding-based multicast network by combining several packets into a single transmission slot.

2.2.5.1 *The Multicast Network Model*

*Anetwork model is* a vivid description of how a set of network layers interact. A network shown in Figure 2.7 is usually represented as a weighted graph G (*V, E*), where *V* is a set of nodes and *E* is a set of directed links, V $G$ and E $G$ denote the number of nodes and links in the network.

As shown in Figure 2.7, the nodes in set V can be of the following three types (source node, intermediate nodes and destination nodes) and they are all called the nodes of the multicast group $V_m$. In other to improve the utilization of the network, the random linear network coding is assumed to be used in the multicast flows, and network coding allows the flows of a multicast session to share network capacity by coding them together (Zhao *et al.*, 2014).

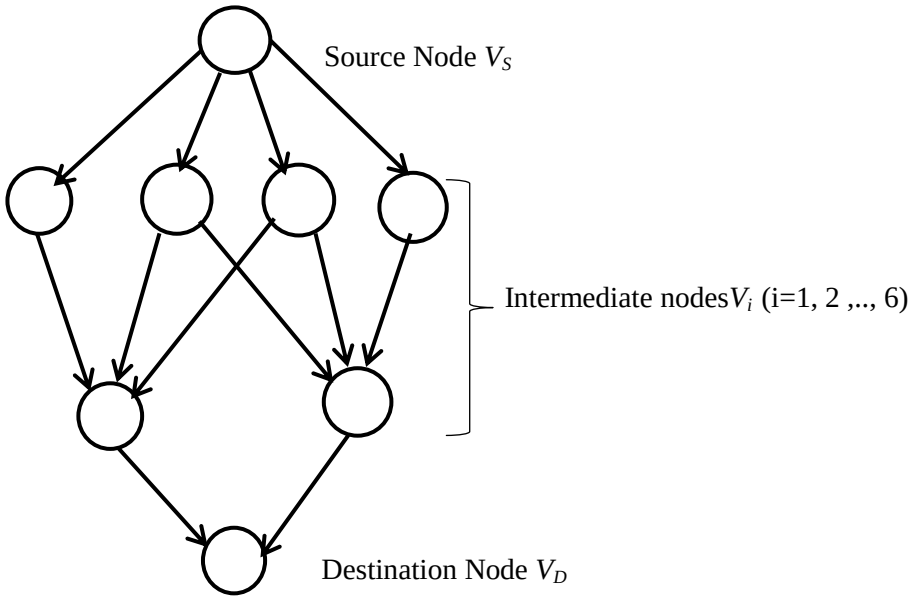



Figure 2.7: A Network Model(Zhansong, 2005)

As shown in Figure 2.7, other nodes are connected to the source that sends out the data stream. Relay node also called the intermediate node is an intermediate hop in the path from the source to a destination while the destination node is connecting a

destination that receives the data stream and they are usually denoted in the multicast tree as $D \subseteq V \ T$

Each link is associated with parameters such as link delay, link cost, link congestion, etc., which describe the current status of the link. Similarly, each node is associated with parameters such as buffer space available, interference which are termed as node state (Dutta & Mishra, 2012). Let $M \subset V$ be a set of nodes involved in a group communication and M is called multicast group with each node $v \in M$ as a group member. Packets originating from a source node $v_S$ have to be delivered to a set of receivers nodes M $v_S$ . Each node *v* is assigned a cost $\gamma \ v$ and a delay $\delta \ v$ which are assumed to be nonnegative integers.

For a node $i$ of the network, the cost of bandwidth required to reach another node $j$ is given by(Swapna *et al.*, 2013):

$$y_{ij} = c_{ij} \tag{2.2}$$

where

$C_{ij}$ is the cost of bandwidth used and $y_{ij}$ is the distance from the source node to destination nodes.

Given a set of nodes n and a cost $C_{ij}$ associated with node $i$ to $j$ $(1 \le i \ne j \le n)$. To find the minimum cost of the sub graph G denoted by $\gamma \ G$ is defined as (Ackermann *et al.*, 2008)

$$\gamma \ G \ = \sum_{v \in E \ H} \gamma \ v \tag{2.3}$$

The delay of a path P in Graph G, denoted by $\delta \ P$ is defined as (Ackermann *et al.*, 2008):

$$\delta\ P\ = \sum_{v \in E\ P} \delta\ i \qquad (2.4)$$

where $\delta_i$ is the delay encountered by the packets $i$ on a given link during multicast

A vertex *s* is designated as the source and a set of D $d_1, d_2, ...d_n$ vertex is designated as the destinations. A tree T is called multicast tree if all the source and destination nodes are all members of the multicast tree T, thus from(Leggieri *et al.*, 2010):

$$s\ \cup D \subseteq V\ T \qquad (2.5)$$

The delay-constrained minimum cost multicast tree problem is to construct a multicast tree T such that $\delta\ T\ \leq \Delta$ and $\gamma\ T$ is minimized, where $\Delta$ is a positive integer (Leggieri *et al.*, 2010).

Minimizing cost of multicast has been widely studied within the frame work of linear programming and the objectives functions are usually defined in the form of minimizing the cost of a multicast subjected to some sets of constraints where the cost is the bandwidth used during multicast. Tree constrained link optimization problems are mostly used where a tree constrained is imposed and a link optimization function is used to locate an optimal multicast tree that fulfills the constraints (Afolabi *et al.*, 2013). Such constraints includes delay, packet utilization, packet loss and inter-receiver delay jitter bound defined as the difference between the end-to-end delay along the paths from the same source to any two different receivers.

multicast algorithm are defined and they can be stated as: given a source node *s*, a set *R* of destination nodes, a set of constraints *C* and the optimization goal is to find the best feasible multicast tree covering *s* and all nodes in *R* which satisfies *C*(Xiong & Li, 2010).

### 2.2.6Performance Metrics that affects Multicast

There are many parameters that hinder smooth and efficient group communication over various networks and some of these performance metrics are:

i. **Packet delay**: The delay of a packet is defined as the time it takes for the packet to reach all its destination after it arrives at its source (J. Liu *et al.*, 2012). The point to point delay measures the one way delay experience by packet from the beginning to the end across a given network (Choi *et al.*, 2007). Packet delay can be summarized into four components; (a) the propagation delay which is determined by physical characteristics of path a packet traverses; (b) transmission delay which is a function of the link capacities along the path as well as the packet size, (c) nodal processing delay is the time to examine the packet header and determines the route for the packet; (d) queuing delay depends on the traffic load along the path, and varies in time depending on how the network is used. In addition to traffic load, sometimes anomalous behaviors of individual routers can introduce other factors that cause packet delay (Bang & Chao, 2009). In communication networks, messages sent by a source towards all the members of a multicast group are required to be delivered within a maximum delay**.** Packet delay is the most difficult constituent of network behavior to examine with loss, because the packet either shows up at the receiver or it does not, while with delay there are numerous shadows of possibility and meaning in the time necessary for a packet to arrive (Shrader & Ephremides, 2012). Similarly, one of the principle elements contributing to delay is queuing within the network, which is very important in understanding how network capacities evolve over time (Karthikeyan & Shankar, 2014).

ii. **Packet loss:** During multicast, packets loss at various nodes directly affects the performance of the networks and it occurs due to buffer overflow or error bit. This can be attributed due to failure of one or more transmitted packet to arrive at their destination. Thesepacket losses vary at different wireless stations because losses are distance location dependent (Tang & McKinley, 2003). This causes noticeable effect during multicasting and can produce jitter in the case of videoconferencing and can create frequent gaps in speech received during audio conferencing. Instead of unicasting multiple copies of a video to individual receivers, video multicast transmits one copy of a video to all receivers in a multicast group simultaneously(Z. Liu *et al.*, 2010). However, video multicast over wireless networks has its own challenges. Without a careful design and consideration, a lost video packet may lead to significant degradation in decoded video quality, due to the use of spatial temporal prediction and sophisticated entropy coding. Wireless video multicast system must be able to handle packet losses induced by the underlying wireless route.

iii. **Packet congestion:** This occurs when the incoming packets at the node approaches or exceeds that of the outgoing packet. In such a case, the length of the queue at the node grows indefinitely and hence, the end to end solution for congestion control becomes very complex(Singh & Yadav, 2012). Congestion has been the major reason for packet delay duringmulticasting and several packets are delayed in the network (Karthikeyan & Shankar, 2014). The longer the path a packet has to travel, the higher its possibility of becoming lost due to collision or congestion, especially under high traffic (Nguyen & Xu, 2007). Congestion control regulates the source rates to avoid overwhelming link capacity (Chopra & Mishra, 2012). Multicast flow causes more congestion than

unicast traffic because it is always distributed in large multicast trees(Chen *et al.*, 2012). With the multimedia applications becoming more popular in wireless network, how to enhance network performance for multicast applications requires urgent attention(Zhao *et al.*, 2014).

iv. **Packet Rejection:** This occurs when the incoming packets cross threshold value and as a result of that thosepackets are discarded in an arbitrary manner without putting into consideration the ranking and properties of the packets (Montessoro, 2013). Most of these packets are either connection oriented packets or connectionless oriented packets like data packet, audio and video packet (Chopra & Mishra, 2012).

v. **Throughput**:Throughput is defined as the total amount of data a receiver *r* receives divided by the time between receiving the first packet and the last packet. The average data taken over all the receivers is the average throughput of the multicast group, assuming that each group has one sender(Swapna *et al.*, 2013). Onesalient consideration in the design of a network is the network's ability to efficiently support high throughput multicast applications over wireless links(Kandari & Pandey, 2014). Interference of multiple wireless links is one of the main challenges for designing high throughput in wireless networks(Zhao *et al.*, 2014). To make video multicast service practically attractive, delivering video multicast frames reliably to receivers is a very important problem that requires urgent attention (Shin *et al.*, 2013).

### 2.2.7Shannon-Hartley Channel Capacity Theorem

According to (Shannon, 1949), the bandwidth of a given channel increases as the noise within the channel also increases. The law is named after Claude Shannon and Ralph Hartley. Shannon–Hartley equation relates the maximum capacity that can be

achieved over a given channel with certain noise characteristics and bandwidth. It can be expressed by the following relationship.

$$C = B\log_2(1+S/N) \quad (2.6)$$

where B is the bandwidth of the channel in Hz, S is the signal power in watts, N is the total noise of the channels in watts, C is channel capacity and S/N is assumed to be noise power.

The bandwidth B is directly proportional to the noise N and hence, considering more parameters leads to an increase in the cost of bandwidth within the channel (Shannon, 1948).

### 2.2.8Review of the Existing Network Coding Algorithm

Networkcoding algorithm was developed base on the principle of network coding aimed at addressing the problem of cost (energy consumption and bandwidth) during multicast. It was proposed by Ajibesin *et al.*, (2012) and it has been used for solving multicast problem by reducing multicast to polynomial time solvable optimization problem. Network coding offers many advantages such as minimizing energy per bits, minimizing delay, maximizing throughput, providing security and reliability. Network coding algorithm was developed within the framework of linear programming and it is mostly considered for network coding of multicast network and it is coded in C programming language.

Performance metrics such as packet delay and congestion control were mostly considered by researchers during multicast (Chen *et al.*, 2012; Xi & Yeh, 2010, 2011; Xiong & Li, 2010; Yang & Zhao, 2014). Some researchers developed more interest to multicast problems associated with delay constraint problems only without considering other parameters which are equally important in optimizing cost. Ajibesin *et al.,* (2012) considered packet delay as the only key performance metric in

developing network coding algorithm. The NCA was developed to bring improvement in only packet delay and the flow chart of NCAis shown in Figure 3.1.

The sequential step of the Network Coding Algorithm does the following:

i. Read as input the test file generated by Multicast Incremental Power Algorithm.
ii. Extracts the associate weight of the network topology.
iii. Performs wireless optimization based on the constraint and solve the problem.
iv. Output the cost as standard output.

However, because network coding extends the functionality of network nodes from storing/forwarding packets to performing algebraic operations on received data,this resulted in the complexity of packet management and increased delay for the coding nodes within the network. Packets delay was the focus of the problem they largely addressed and it was considered as the only key performance parameter that brings improvement in the development of the algorithm.The NCA was coded in C-programming language and the problem was formulated as an integer linear programming problem. The NCA is shown in appendix D.

### 2.2.9The Improved Network Coding Algorithm

The improved network coding algorithm is an extension of work carried out by Ajibesin *etal.,* (2012) and Ajibesin *et al.,* (2013) where they considered packet delay as the only key performance metrics whereas there are other parameters such as packet loss, packet congestion, delay jitter, packet rejection and throughput that are equally important. Considering packet loss, packet rejection and packet delay leads to the improvement of what the NCA did.The INCA was proposed in such a way that it can be applied to two or more performance metrics collectively. In communication

networks, messages sent by a source towards all the members of a multicast group are required to be delivered within a maximum delay.The value of packet delay was chosen to be 400ms in order to agree with that of the benchmark algorithm (NCA), to enhance efficient packet delivery and to take care of large variations in end to end delay (Ajibesin *et al.*, 2012; Karthikeyan & Shankar, 2014). This also takes care of other delays such as nodal processing delay, which deals with the physical characteristics of the path a packet traversed. Delay on the process of transmission which is a function of link capacity, delay along the route, and queuing delay which depend on the traffic load along the path. These delay lead to bandwidth consumption during multicast and on the other hand, packet loss is also critical because when sending packets from the source to receivers, there are cost associated with them, and if they do not get to their destinations, there is additional cost involved in resending the packets again. So, a negligible value of the packet loss is set to be equal to four (4) in such a way that the cost of lost packets within this range is tolerated(Sommers *et al.*, 2005). According to Frost and Stewart (2011), missing packets less than four (4) every now and then will not affect the quality of the conversation. The value of rejected packets was also chosen to be 8 to avoid retransmission of rejected packets which are discarded as they reach their threshold value (Chopra & Mishra, 2012).

## 2.3 REVIEW OF SIMILAR WORKS

Considerable amount of work has been dedicated to minimizing cost of multicast over coded packet wireless networks. The key ones reviewed are as follows:

**Ahlswede *et al.*, (2000)** introduced new class of problems called information flow rate were they referred coding at a node in a network as network coding. Their main findings was focused on employing the use of network coding to achieve optimality

rather than just considering the information to be multicast as physical entity. They also showed that optimality can be achieved if two information sources are independent and are coded by superposition. They based their discussion on a class of block codes called α- codes and the results obtained was regarded as the max-flow minimum cut theorem for network information flow. However, the results obtained for multisource is most likely not applicable to general coding schemes.

**Sanders *et al.*, (2003)** extended the famous max flow min-cut theorem of network coding proposed by Ahlswede*et al.,* (2000)by employing the use of polynomial time algorithm for constructing coding schemes for multicasting at max data rate. The algorithm adopted the use of random linear network coding in acyclic network rather than α class of block codes. However, acyclic network is not applicable to real time situation most especially when the rate of flow is not too large with the network.

**Lun, *et al.*, (2006)** worked on minimum cost of multicast over coded packet network. They provided an overview of minimizing the total cost of energy consumption for multicast packets in wired and wireless networks. They established minimum cost connections by first determining the rate to inject coded packets on each arc, and then determined the contents of those packets. They achieved reduction in the problem of multicast programming. However, some of their limitations were the stability of their decentralized algorithm under changing conditions and exploring a specific approximation method for use in the formulation of dynamic multicast.

**Bhadra, *et al.*, (2006)** worked on a single source minimum cost problem which was framed as a convex optimization problem with the use of network coding and convex increasing edge cost. They analyzed the cost for the scenario where each of the multicast receivers routed its flows. They constructed a selfish flow steering

algorithms for each receiver and the edge cost at each edge was allocated. However, encoding /sending of packet to different sink were restricted because the algorithm was constructed at each receiver.

**Zongpeng Li, (2008)**examined cross monotonic multicast by concentrating on three things namely, optimal flow routing (keeping the routing cost to be shared as low as possible), cross monotonic cost sharing and budget balancing (recovering the flow cost from user payments). He found that no cross monotonic multicast could satisfy all the three conditions. Cross-monotonic cost sharing was a situation where each user's payment smoothly decreased as the service set expanded.The aim of the research was to achieve strategy proof in both directed and undirected network. However, constant cost was infeasible in directed network.

**Xiong and Li, (2010)** worked on minimum cost optimization of multicast wireless networks. Their research focused on an information model that differentiated intermediate nodes in a multicast network with network coding, routing or replicating nodes. They developed a means of solving the minimum energy cost multicast problem in wireless networks by optimizing transmission energy, the number of network coding nodes as node classification and the number of packets undergoing network coding. The simulations results showed that the proposed method was good at achieving a minimum cost but very complicated for many multimedia applications.

**Li and Chu (2012)** developed a multicast strategy proof where no receiver had incentive to lie about its true valuation and the cost incurred. They also designed multicast schemes that pursued maximum budget balance and routing optimality. However, the multicast strategy proof that pursued maximum budget balance and routing optimality could not be applicable to users of larger group within the network

because the cost incurred by each receiver could not be properly ascertain with users of larger group.

**Tan, *et al.*, (2012)** used genetic algorithm to minimize coding nodes in wireless networks. They considered single source and single destination in their frame work and they proved that Network coding with genetic algorithm improved the throughput for wired and wireless network. The simulation results also showed that the average end to end delay of the network after minimizing coding nodes was less than the normal network. However, there was complexity of packets management and increased delay for the coding nodes.

**Chen and Ho (2012)** studied congestion control for Multicast Flows with network coding. The problem was formulated as an optimization based model for network resource allocation base on two sets of decentralized controllers as source and intermediate nodes for congestion control. They presented two models for congestion control, one for networks with given coding sub-graphs and one where such sub-graphs are found dynamically. The aim of the research was to carry out congestion control by adjusting the flow rate on each tree. Each source node adjusts its sending rate according to the local congestion price to avoid communication overhead for congestion control within the network. However, the stability of their controller under propagation delay was a great challenge.

**Ajibesin, *et al.*, (2012)** studied the performance of multicast algorithms over coded packet wireless network by using Network Coding Algorithm (NCA) and Multicast Incremental Power Algorithm (MIPA). They were particularly interested in showing the performance of network coding on multicast network when the network arcs were weighted. The result showed that network coding based-algorithm out-performed

multicast incremental power algorithm. However, the network coding based-algorithm was developed using linear programming framework and they considered only packet delay as a performance metric that affected multicast (constraint) in developing the algorithm.

**Ajibesin, *et al.*, (2013)** worked on cost-efficient multicast over coded packet wireless networks using Data Envelopment Analysis (DEA) by first demonstrating the fundamental differences between Network Coding Algorithm (NCA) and Multicast Incremental Power algorithm (MIPA). The (DEA) is operational research tool widely used to determine the effectiveness and efficiency of any organization, algorithm etc. in terms of cost, performance and service delivery. It consists of entities called the Decision making units (DMUs) for evaluating the performances of the two algorithms. This was followed by the analysis of the statistical mean of the multicast cost to determine the cost-effectiveness and cost-efficiency of the two algorithms. Their findings showed that NCA outperformed MIPA for both cost-effective and cost-efficient multicast. However, performance parameterswhich affect multicast traffic during an event such as packet congestion, lost, rejection were not considered which was a great limitation in achieving efficient data delivery during multicast.

**Rajkumar, (2014)** proposed an efficient resource allocation in multicasting over mobile ad hoc networks by choosing the closest distance from the source to any node as zone leader to communicate with its group members. Only zone leader transmitted data to other nodes. The proposed model used geographic forwarding to provide reliable packets transmission. However, some large numbers of packets were missing in the process of delivery and hence complete packets cannot get to the destination nodes.

**Mani, (2014)** examined secured multicasting for wireless sensor networks. He used public key cryptography for authentication using signature remuneration that overcomes the vulnerability of symmetric based schemes. He made use of ecliptic curve digital signature algorithm to ensure confidentiality and the integrity of the messages exchanged during multicast as against the symmetric schemes; Hash-based Message Authentication Code (HMAC) and RivestShamirAdlemanused by other researchers. The scheme was designed to offer three levels of security during multicast and to meet low overhead, immediate authentication, strong authenticity resilience to node compromise attack. However, the scheme designed was only compared with only two public key cryptography scheme and can be more vulnerable to hackers.

**Kandari and Pandey, (2014)** examined quality of service multicasting in Mobile Area Network (MANET). They considered two techniques in phases. Phase one selects all paths satisfying quality of service constraints and phase two selected the best path for building the multicast tree. These techniques took care of quality of service constraints such as bandwidth, delay, packet loss and jitter. The scheme also ensured access control, data integrity and group confidentiality. However, quality of service, power control, multiple source and address configuration were not taken care of.

**Lagercrantz, (2014)** studied multicasting in an online gaming system. He focused on sets of interconnected multicast managers for receiving messages from game servers to the end-user. One multicast manager received each message and its associated end-user recipient with one multicast server been connected to each multicast manager. There existed a memory for storing information associated with each end user with a multicast manager or server pair to which it was been connected. The game servers

were connected to the multicast managers through a load balancer. However, there was scalability problem that arose when trying to multicast large amount of data to many end-users in real time.

**Yin, et al., (2014)**studied the bounding advantage of multicast network coding in general network. They constructed two classical types of parameterized networks (bi-directed and hyper network) in order to study the potential benefits of network coding in terms of increasing throughput, security and reducing transmission cost. The aim of the study was to find out how large can the network coding advantage be in directed and hyper networks compared to directed and undirected networks. Besides other advantages, they showed that network coding has less over all bandwidth consumption to achieve a targeted multicast group. However, the parameterized networks considered are not applicable to real time situation.

**Zhao et al., (2014)** studied the convergence of decentralized min-cost sub graph algorithm for multicast in coded networks. They developed an algorithm that generates a feasible solution after each iteration to take care of the dual sub gradient method proposed by Lun *et al*, (2006). They completely decentralized the system as there is no iteration between the nodes and each node only knows the cost of its incoming and outgoing links. However, the algorithm did not take care of interference within the wireless network and hence the need to extend the algorithm to interference-limited wireless network is a great step toward achieving efficient result.

Based on the reviewed literatures and their shortcomings, it is evident from the literatures that minimizing cost during multicast has been given a significant attention by various researchers and it is on this evidence that this research is motivated. There is still be need to improve the performance of the network coding based-algorithm in

order to investigate Shannon Hartley Channel Capacity theorem by considering more performance metrics over coded packet wireless network. This research is an extension of the work carried out by Ajibesin *et al.*, (2012) and Ajibesin *et al.*,(2013) were they considered only packet delay where as there are other performance parameters that are equally important. The Network Coding Algorithm that they developed can only bring about the improvement in packet delay but the INCA can be applied to two and three performance metrics (packet loss, packet delay and packet rejection) collectively. Their research is based on parameter specification, for example, numbers of randomly generated nodes were specified as (20,30,40,50, 60, and 70) and the number of receivers were also specified (2,3,4,5,6,7,8,9,10).

# CHAPTER THREE
# METHODS AND MATERIALS

## 3.1 INTRODUCTION

This section explains the detailed procedure and steps carried out to improve the performance of NCA where two and three performance metrics were considered separatelyin other toinvestigate the problem as proposed by Shannon in 1948.

## 3.2 Mathematical Models for Minimum Cost of Multicast

The minimum-cost multicast can be modeled in terms of graph by considering the devices of the network as nodes and transmission links as arcs. Let G (V, A) be a directed complete graph where V, represent the set of the terminals of the network and A is the set of directed arcs which connect all the possible pair $i, j$ with $i, j \in V$ and $j \neq i$. Each intermediate or destination node $i \in V$ can receive data from any other nodes of the network and send data to any node in its transmission range. Let $s \in V$ be the source node of the message and a subset of nodes $R \subset V$ whose element are the destination of the communication. It is assumed that the nodes are fixed since static network are being considered and all the distances $d_{ij}$ between each pair of nodes $i$ and $j$ (source, intermediate and destination nodes) in V are noted for real world applications (Schalkamp, 2012). With each arc $i, j$ is associated a cost $c_{ij}$ that represent the minimum cost required to establish a directed connection to node $i$ to node j. The range assignment function assigns each node $i \in V$ its transmitting cost $c\ i$ (Bosch & Trick, 2004).

C=$V \rightarrow \square^{+}$, $i \mapsto c\ i$ (3.1)

where $\square^+$ is the range assignment function, V is the node within the network and $c\ i$ defines the cost of bandwidth to be optimized.

These metrics namely delay, rejection and loss of packets are represented by a range assignment function as (Zhansong, 2005):

Delay $\delta(i) : \mathrm{V} \rightarrow \square^+$

Packet loss $pl\ \ i\ : \mathrm{V} \rightarrow \square^+$

Packet rejection $pr\ \ i\ : \mathrm{V} \rightarrow \square^+$

$\delta\ \ i\ ,\ \ pr\ \ i$ and $Pl\ \ i$ define the criteria to be bounded(Byrka & Grandoni, 2010).

### 3.2.1Cost Optimization in Coded Packet Wireless Networks

In coded packet networks, nodes are able to perform some functions on multiple incoming packets like $\mathrm{A} \otimes \mathrm{B}$, and the coupling constraint changes chronologically. For a given network graph G denoted by $(V_G, E_G)$, where $V_G$ is the set of nodes in the network and $E_G$ is the set of edges which connect the nodes of $V_G$. The set $Vm$ is divided into three subsets $V_S, V_D$ and $V_I$ (Avni *et al.*, 2014) and is given by:

$$V_M = V_S \cup V_D \qquad (3.2)$$

where $\mathrm{V_M V_I}$, $\mathrm{V_S}$and $\mathrm{V_D}$ are the multicast group, source node, intermediate node and destination node within the network.

The intermediate nodes are always used for relaying multicast traffic, but do not belong to the multicast group. Mostly, $V_G$ are routers within the multicast group and each edge $e \in E_G$ has a positive cost value C(e) that measures the quality of the edge. The corresponding cost is denoted by $\infty$ if there is no edge between the nodes. The

optimization objective of the problem is to find a tree (Tayu *et al.*, 2005) that certifies this expression:

$$T = (V_M + V_\theta + E_\theta) \quad (3.3)$$

where T, $V_\theta, E_\theta$ and $V_M$ are the multicast tree, the subset of the intermediate nodes, edges in the network and the multicast node that minimize the cost for connecting the nodes in $V_M$ and that satisfies the multicast constraints. $V_M$ is the multicast node.

### 3.2.2 **Mixed Integer Linear Programming Formulation**

The problem can be formulated as a mixed integer programming where the optimization objectives are usually defined in terms of minimizing the cost of a multicast tree, and the cost in question is the bandwidth used, loss, delay and rejection of packets serves as the constraints. The objective function is stated as:

$$\text{Min (q)} = \sum_{(i,j)\in V}^{n} c_{ij} \qquad \forall (i.j) \in V, i \neq j \quad (3.4)$$

Subjected to:

$$C_{ij} \leq C_{ia_j^i} \qquad \forall (i,j) \in A, a_j^i \neq i \quad (3.5)$$

$$\delta(p) \leq 400 \qquad \forall (i.j) \in V, i \neq j \quad (3.6)$$

$$\rho(l) \leq 4;\ i = \text{Source},\ i \neq j \qquad \forall (i,j) \in V \quad (3.7)$$

$$p(r) \leq 8;\ i = \text{Source},\ i \neq j \qquad \forall (i,j) \in V \quad (3.8)$$

$$\sum_{y=1}^{n} y_{ij} > 0\ ;\ i = \text{Source},\ i \neq j \qquad \forall i \in v/\ s \quad (3.9)$$

where $C_{ij}$ = the cost of the bandwidth for sending messages from source node to destination nodes during multicast

$y_{ij}$ = the distance from source node to destination nodes.

$(i, a_j^i)$ = the arc originating in node *i* with the lowest cost.

q= is the objective function

*pd* = packet delay

*pl*= packet loss

*pr* = packet rejection

In formulating the problem, it is assume that the number of loss packets isless than four (4) decibel which represents5.7% for the highest number of randomly generated nodes in order to agree with other values established in literatures (Mwela, 2010; Perkins *et al.*, 1998), packets delay isless than 400ms(Ajibesin *et al.*, 2012; Deng, 1995)and the number of rejected or dropped packets to be less than eight (8) which represent 11.42% of the highest number of randomly generated nodes. In the work of Maher *et al.*, (1998), the number of rejected packet represent 8% of the total packet sent.

Most multimedia applications such as video and audio conferencing have much higher delay allowance of several seconds. The choice of 400ms is prompted to allow maximum flow of packets during multicast, to ensure efficient packet delivery and to take care of large variations in end to end delay in order to agree with values existed in literatures (Ajibesin *et al.*, 2012; Rhee, 1998). This also takes care of other delays such nodal processing delay, which deals with the physical characteristics of the path a packets traversed, delay on the process of transmission which is a function of link capacity, delay along the route, and queuing delay which depend on the traffic load along the path(Guan *et al.*, 2012; Zhang *et al.*, 2011).

In formulating the problem, the value of packet loss was chosen to be 4 decibels which is equal to 5.7% to avoid complete retransmission of packet and degrading of video quality because video are synchronize with voice during multicast (Rhee, 1998).These values are expressed in percentage of the packet sent for a fixed number of nodes within the multicast group (Minhas *et al.*, 2012). This value was not

expressed in percentage as other literature does because different numbers of randomly generated nodes were used for the simulation (Ma & Denko, 2008). Packet loss reduces the throughput of the network and increase latency due to additional time needed for retransmission. A single occurrence of packet loss can introduce error in a reference frames which can propagate to its succeeding frames and get amplified as more packet are lost. The role of a network device is to ensure that packets are forwarded as efficiently as possible within a limited time.

The constraint in equation (3.9) means that the distance from the source node through the destination nodes is greater than zero.

Similarly, Packet rejection or dropping occurs when queues are not efficiently managed and packets begin to experience delay in such a way that the amount of buffered packets is greater than the queue length(Anna *et al.*, 2004). Packets that are delayed too long are rejected in order to meet the timing requirement. Packet rejection also occurs when packets are corrupted on transit and when the buffer occupancy is above the threshold value, the arriving low priority packet will be rejected (Dimitri Bertsekas & Gallager, 2001). Most numbers of rejected or dropped packets existed in literatures are expressed in percentage of the packets sent(Maher *et al.*, 1998).The value of rejected or dropped packets was assumed to be 8 decibel which represent 11.42% to avoid complete buffer occupancy and lip synchronization between the audio and video during multicast(Dimitri Bertsekas & Gallager, 2001).This value was not expressed in percentage of the packets sent because the problem was formulated as mixed integer linear programming.

Since the aim of the research is the development of improved multicast algorithm for the determinationof the cost of bandwidth within the channelin order to interpret Shannon Hartley capacity theorem, it is imperative to note that there is cost associated

with sending packets from the source node to various destination nodes (receivers). Sometimes those sent packets are lost and rejected in the process of reaching their destination andan additionalcost of resending themagainis requires. The constraints take care of any value of packets lost that is less than four (4) during multicast.In a situation where the value of packet lost is greater than four, the output solver will resend those packets in orderto form another wireless constrain based on the problem formulation described in equations (2.9) to (2.14). Similarly, there is no any acceptable value for packet delay available in literatures. For example, researchers(Ajibesin *et al.*, 2012; Jean-Chrysostome Bolot, 1993; Jean-Chrysotome Bolot, 1993; Choi *et al.*, 2007; Douik *et al.*, 2014; Frost & Bryant, 2011; J-C, 1993; Karthikeyan & Shankar, 2014; Wu *et al.*, 2014; Zhu *et al.*, 1995)all adopted different values for packet delay for minimum cost of multicast aimed at achieving optimal solution.

The Improved Network Coding Algorithm is represented in Appendix A-C and the sequential steps are as follows:

**i.** The algorithm read as input the randomly generated nodes from the MIPA. The sniped portion of the algorithm shows the listing program that reads the file from the input.

```
void filetest(char *filename, double **data_ptr, int *num_edges)
{
  // Variables to extract information about a single link.
  int out, in, DEBUG =0;         // out = starting node; in = ending node;
  double lat;                    // lat = link distance;
  int line_cnt=0, i = 0;

  FILE *file_in;                 // internal variable assigned to input file
// Checking for the total number of edges and passing the address of data to be stored
  double *M = (double *)malloc(line_cnt*3*sizeof(double));
  file_in = fopen(filename, "r");
  if(file_in ==NULL){puts("There is no such file"); return;};
```

**ii.** The INCA code is applied to the file to test the data stream.

iii. The algorithm forms wireless constraints formulated as mixed integer linear programming problem base on the problem described in the menu scripts. The sniped portion of the algorithm shows the problem formulation.

```
/* Constraint Matrix:
     t: number of sinks,
     e: number of edges;
     v_cnt: number of nodes;
         pd: packet delay, (≤400)
         pl: packet loss; (≤4)
         pr: packet rejection (≤8)
     |A_z| |X_z| = |b|
     |A_x| |X_x|          */
  /* Defining A_z Matrix (e(t+1) by et)
     Represents the additional variable constraint:
     z_e - x_e^t != 0*/
  //int *A_z = (int *) calloc((2e+1)*e*t*3*400*8, sizeof(int)); // e*(e+2)*t*4 elements; 3
entries for each
  int inx = 0, cnt=0;
  if(wireless)
  {
   printf("\nPerforming wireless optimization: \n");
   for(j=0;j<t*e;j=j+t)
    {
        int temp[2];         // collect all indices with cost >= A[cnt]
```

iv. Formulation of an A_Z matrix as an array and the output is sent to GNU Linear programming solver. The sniped portion of the algorithm shows the formation of the problem in form of matrix.

```
sprintf(var_name, "B%i", i);
      glp_set_row_name(lp, i, var_name);
      glp_set_row_bnds(lp, i, GLP_FX, (double) *(B+i-1), (double) *(B+i-1));
    };

  printf("\n");
  num_cols = (int)2*(e*t)+e*400;
  glp_add_cols(lp, num_cols);
  for(i=1;i<=e;i++)
    {
      sprintf(var_name, "x%i", i);
      glp_set_col_name(lp, i, var_name);
      glp_set_col_bnds(lp, i, GLP_DB, (double) 0.0, (double) *(C+i-1));
      glp_set_obj_coef(lp, i, (double) *(A+(i-1)));
      // printf("%s ", var_name);
    }
  for(i=e+1;i<=e*t+e;i++)
    {
      sprintf(var_name, "x%i", i);
      glp_set_col_name(lp, i, var_name);
      glp_set_col_bnds(lp, i, GLP_LO, 0.0, 0.0);
      glp_set_obj_coef(lp, i, (double) *(A+(i-1)));
```

v. The cost is computed using simplex method and it is displayed as standard output. The sniped portion of the algorithm shows the cost of multicast for groups of receivers represented by X2…, X10.

```
  glp_load_matrix(lp, ne, ia, ja, ar);

  glp_simplex(lp, NULL);

  Z = glp_get_obj_val(lp);
  x1= glp_get_col_prim(lp, 1);
  x2= glp_get_col_prim(lp, 2);
  x3= glp_get_col_prim(lp, 3);
  x4= glp_get_col_prim(lp, 4);
  x5= glp_get_col_prim(lp, 5);
  x6= glp_get_col_prim(lp, 6);
  x7= glp_get_col_prim(lp, 7);
  x8= glp_get_col_prim(lp, 8);
  x9= glp_get_col_prim(lp, 9);
  x10= glp_get_col_prim(lp, 10);

  printf("Z= %f; x1 = %f x2 = %f x3 = %f x4 = %f x5=%f \n", (double) Z, (double)x1,
(double)x2, (double)x3, (double)x4, (double)x5);
    printf("x6 = %f x7 = %f x8 = %f x9 = %f x10= %f\n", (double)x6 ,(double)x7, (double)x8,
(double)x9, (double)x10);
    glp_delete_prob(lp);
```

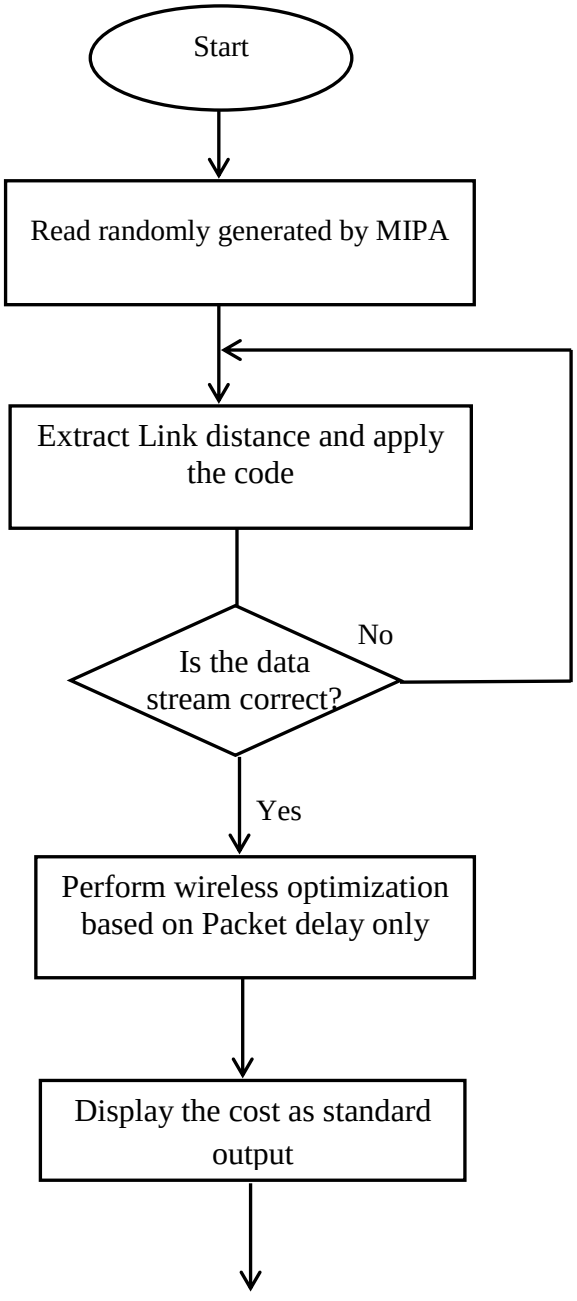
The operation of the flow chart shown in Figure 3.1 is explained thus;

The algorithm reads as input the total number of randomly generated nodes by the Multicast Incremental Power Algorithm (MIPA) based on the command argument of the MIPA. This is then output to the test file and the test file uses the algorithm code to verify the data stream. If it is not in conformity with the code, this indicates that there is a problem of reading the input data. If the number of the randomly generated nodes by the MIPA is correct, Formulate.1 forms the wireless constraints based on the description of problem formulation stated in this research work as mixed integer linear programming.

The distances of the nodes together with the constraints forms an A_Z matrix as an array and sends the output to Sim1. The array contains relevant information about the nodes (distances and location). The Sim1 sends all this information to GNU Linear Programming solver where the cost is computed. If the output solver is not correct, the routine will move back and form another A_Z matrix based on the wireless

constraints and resend the information to Sim1. However, if the solver output is correct, it displays the cost. This process continues based on the command argument for different number of nodes generated to different receivers within the multicast group.

The flow chart of the NCA and the INCA are shown in Figures 3.1 and 3.2 respectively. Looking at the flow charts, the difference between the INCA and the NCA is bolded in black color which is not the case in NCA.



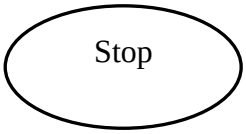
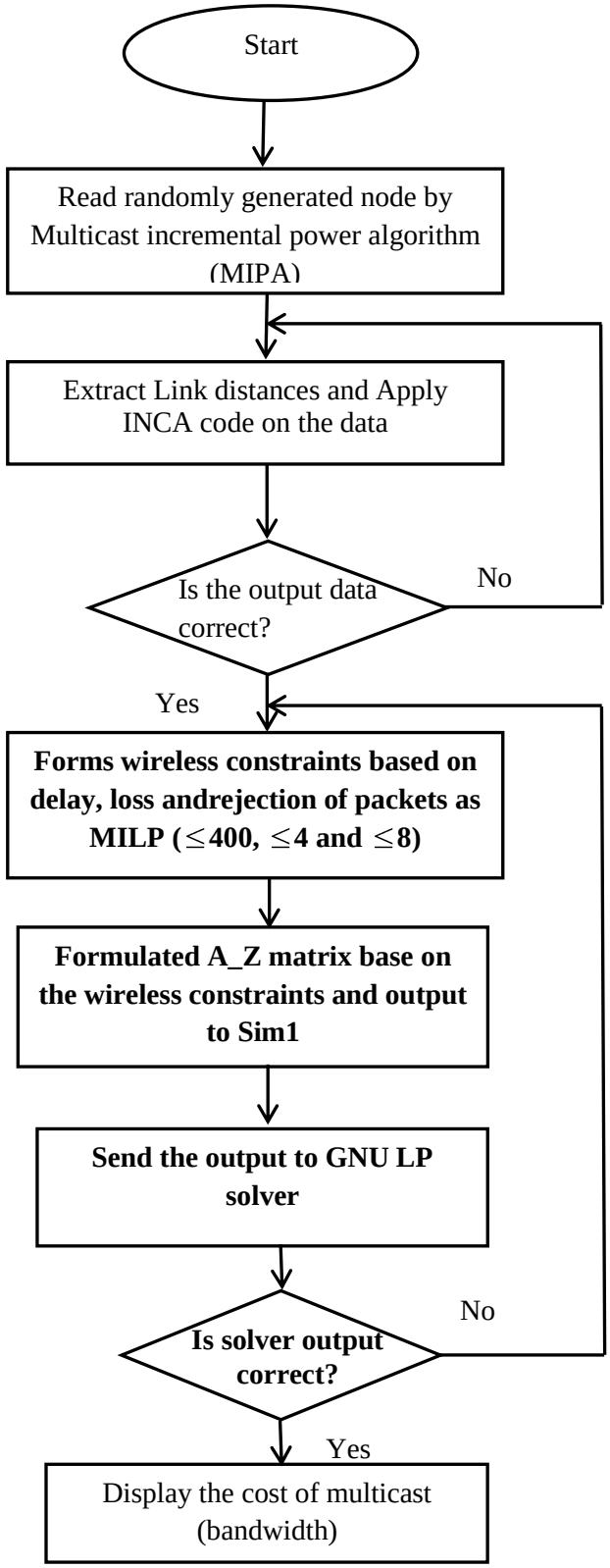



Figure 3.1: Flow Chart of Network Coding Algorithm (Ajibesin *et al*, 2012)



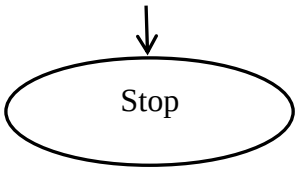



Figure 3.2: Flow Chart of the Developed Algorithm(INCA).

## 3.3 COMPARATIVE ANALYSES OF NCA AND THE INCA.

The salient differences between the two algorithms are as follows:

i. The NCA in the work of Ajibesin *et al.,* (2012) can only bring about the improvement in packet delay whereas in the case of INCA both packet loss ,packet delay and packet rejection can be applied collectively to bring improvement.

ii. The NCA cannot be applied to any other performance metrics except Packet delay while the INCA can be applied to two and three performance metrics collectively.

iii. The NCA was formulated based on delay constrained problem using linear programming framework whereas the INCA with two and three performance metrics was formulated as mixed integer programming problem.

iv. In the case of NCA, the formation of an A_Z Matrix was absent because the algorithm was developed using packet delay only. However, because more than one performance metrics were considered, there is a formation of an A_Z matrix. This is clearly bolded in the flow chart of the INCA.

v. There was no GNU linear programming Solver in the case of NCA whereas in the case of INCA, it was present because of the Formation of an A_Z matrix and the GNU LP solver solves the problem using simplex method which was also indicated with red color in the flow chart of INCA.

The relationship between the number of receivers and the cost of bandwidth for both the NCA and the INCA is given by:

$$R = x(m, n) \tag{3.10}$$

where R is the number of receivers, x is the intercept on y-axis, m is the cost of bandwidth for NCA and n is the cost of bandwidth for the INCA.

The receivers R are the independent variable and both the NCA and INCA serves as dependent variable and they allow researcher to maintain control, measure results and draw accurate conclusions.

## 3.4 INSTALLATION AND CONFIGURATION

Visual studio 2010 was installed along with compilers and relevant libraries for Java and the C languages. The C compiler, GCC was also installed which can be obtained from http://gcc.gnu.org/. In addition, the (GLPK) GNU Linear Programming Kit library (available from http://www.gnu.org/software/glpk/glpk.html) was installed, in order to launch the algorithm from within the C program. The GLPK is an open software package that is used to solve mathematical programs specially, linear program via revised simplex method. Primal –dual interior method, mixed integer linear programming via algorithms.

## 3.5 SIMULATION

The Java program was run first by specifying the pathname of the file for the number randomly generated nodes with the specification of the number of receivers. The test file from the java program was saved in a folder and the Network Coding Algorithm (NCA) took as input the number of randomly generated nodes from the test file and output the cost of bandwidth used during multicast.

Similarly, the Improved Network Coding (INCA) was run using the same test file of the randomly generated nodes with the specification of the number of receivers. The INCA performs wireless optimization based on the problem formulation described in this research work and displays the cost. This process was repeated for 30, 40, 50, 60 and 70 randomly generated nodes multicasting to 2, 3, 4, 5, 6, 7, 8, 9, and 10 receivers for both NCA and the INCA.

### 3.5.1 Simulation Parameters

This research is based on parameter specifications and the experimental parameters that were considered include;

a. Network size: This represents the number of random nodes that are generated by the MIPA, which are 20, 30, 40, 50, 60 and 70.
b. Network dimensions: These define the dimension of the multicast network, which is 10 by 10.
c. Performance metrics: These define the performance metrics considered in improving the performance of the algorithm and they are represented by functional inequalities ($\leq$400ms for packet delay, $\leq 4$ for packet loss and $\leq 8$ for packet rejection).
d. Multicast group: this defines the number of receivers used for the whole multicast session, which are 2, 3, 4, 5, 6, 7, 8, 9, and 10.

The flow chart of how the improved network coding algorithm works is shown in Figure 3.2

The decrease in the average cost of bandwidth when the INCA with two performance metrics is compared with the NCA is given by

$$F = \frac{A-B}{A} X\ 100 \qquad (3.11)$$

Where: F is the decrease in the cost of bandwidth during multicast in Mega bytes per second. A is the total cost of bandwidth for the NCA for all the number of randomly generated nodes

B is the total cost of bandwidth for the INCA with two performance parameters for all the number of randomly generated nodes

Similarly, the decrease in the average cost of bandwidth when the INCA with three performance metrics is compared with the NCA is given by

$$F = \frac{A-C}{A} X\ 100 \quad (3.12)$$

Where C is the total cost of bandwidth for the INCA with three performance metrics for all the number of randomly generated nodes.

And lastly, the decrease in the average cost of bandwidth when the INCA with three performance metrics is compared with the INCA with two performance metrics is given by

$$F = \frac{B-C}{B} X\ 100 \quad (3.13)$$

Where F, B and C are defined in equations (3.2) and (3.3)

# CHAPTER FOUR
# RESULTS AND DISCUSSION

## 4.1 INTRODUCTION

This section shows the performance of the Improved Networking Coding Algorithm (INCA) with two and three performance metrics implemented on visual studio 2010. These randomly generated nodes were used in order to agree with the one used by the NCA which serves as a benchmark. The costs of randomly generated nodes (20, 30, 40, 50, 60, and 70) multicasting to various groups of receivers (2, 3, 4, 5, 6, 7, 8, 9, and 10) are obtained.

## 4.2 PERFORMANCE EVALUATION OF THE TWO ALGORITHMS

Tables 4.1 through 4.6summarized the simulation results obtained for the two algorithms that is Network Coding Algorithm where only packet delay is considered and the improved Network Coding Algorithm where packet delay and packet loss were considered. The two algorithms were then compared in terms of their respective cost of bandwidthand the decrease in the average cost of bandwidth is computed using equation 3.2.

Table 4.1: Costsof Bandwidth for 20 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of Receivers | Cost of bandwidth in Mega bytes per seconds (Mbps) | |
|---|---|---|
| | Network coding Algorithm | Improved network coding algorithm |
| 2 | 1.0710 | 0.3405 |
| 3 | 4.0521 | 1.5047 |
| 4 | 3.6518 | 1.6769 |
| 5 | 3.9777 | 2.2611 |
| 6 | 4.0871 | 3.2449 |
| 7 | 5.9811 | 4.9350 |
| 8 | 7.6331 | 5.5823 |
| 9 | 8.3482 | 6.2030 |
| 10 | 8.5710 | 8.4356 |

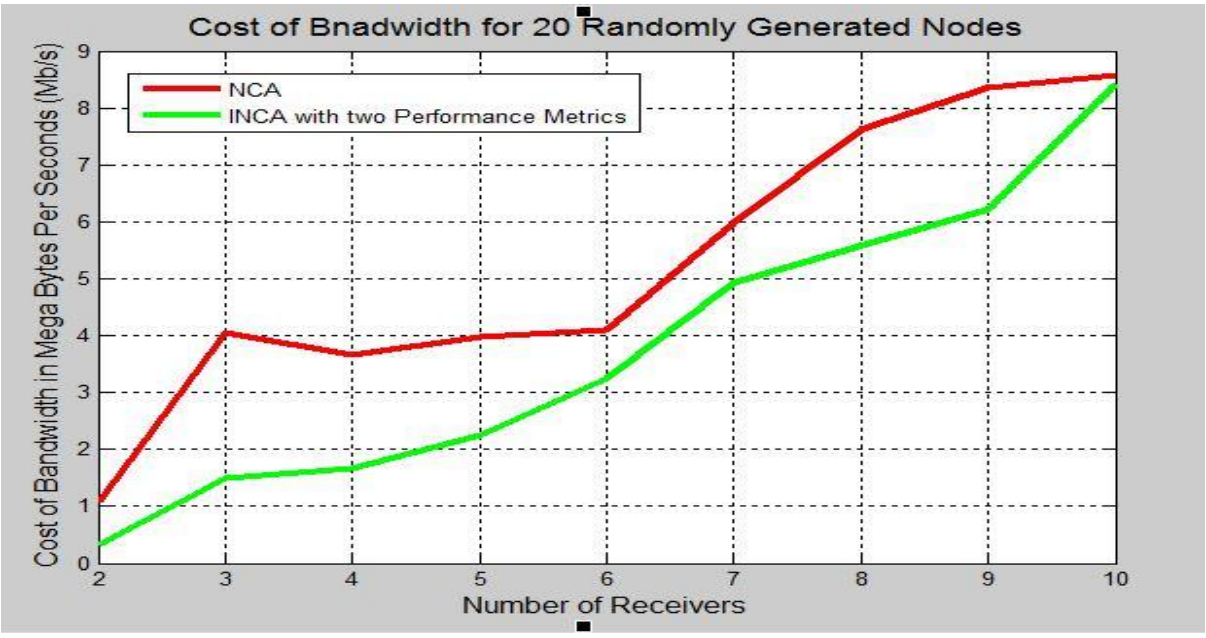



Figure 4.1: Cost of Bandwidth during Multicast for 20 Randomly Generated Nodes.

Figure 4.1 shows the cost of multicasting for 20 randomly generated nodes to groups of receivers for both the INCA and the existing NCA that is used as a benchmark for validation. Simulation results shows that the INCA produces significant reduction in the average cost of bandwidth used during multicast by 27% as compared to NCA.

Table 4.2: Costsof Bandwidth for 30 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of Receivers | Cost of Bandwidth in Mega bytesper seconds (Mbps) | |
|---|---|---|
| | Network coding algorithm | Improved network coding algorithm |
| 2 | 3.6390 | 1.1712 |
| 3 | 4.7158 | 1.9561 |
| 4 | 3.5701 | 2.4960 |
| 5 | 6.9022 | 3.8566 |
| 6 | 4.621 | 2.8661 |
| 7 | 7.2319 | 4.9538 |
| 8 | 6.7638 | 6.3597 |
| 9 | 7.3621 | 6.6312 |
| 10 | 8.9467 | 7.9821 |

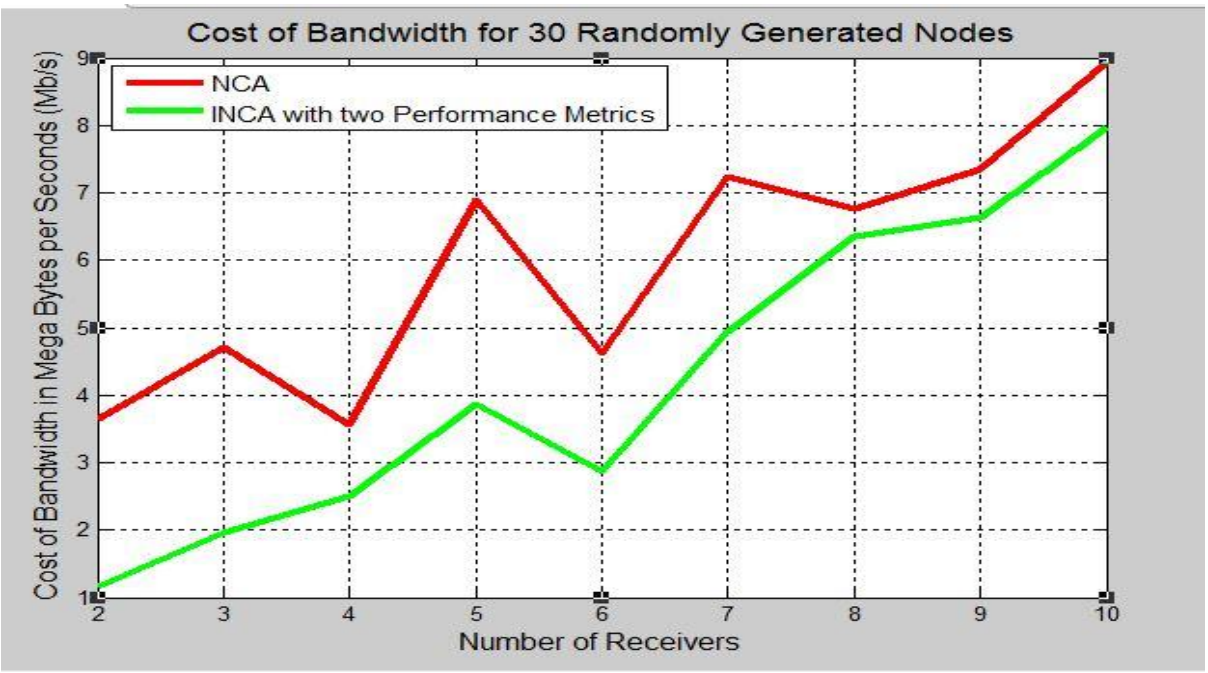



Figure 4.2: Cost of Bandwidth during Multicast for 30 Randomly Generated Nodes.

Figure 4.2 shows the cost of multicast for 30 randomly generated nodes to groups of receivers. Using equation 3.2, the average cost of bandwidth was minimized by 28.7% for 30 randomly generated nodes. The cost of bandwidth when multicasting to five and six groups of receivers rises and decreases sharply because each node has different characteristics.

Table 4.3: Costsof Bandwidth for 40 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of Receivers | Cost of Bandwidth in Mega bytes per second (Mbps) | |
|---|---|---|
| | Network coding algorithm | Improved network coding algorithm |
| 2 | 1.8709 | 0.4333 |
| 3 | 2.3336 | 1.3993 |
| 4 | 3.1568 | 1.8031 |
| 5 | 4.2686 | 2.3311 |
| 6 | 4.7431 | 2.6830 |
| 7 | 4.9952 | 3.6694 |
| 8 | 5.9287 | 4.8068 |
| 9 | 7.4452 | 5.7138 |
| 10 | 8.8209 | 7.8734 |

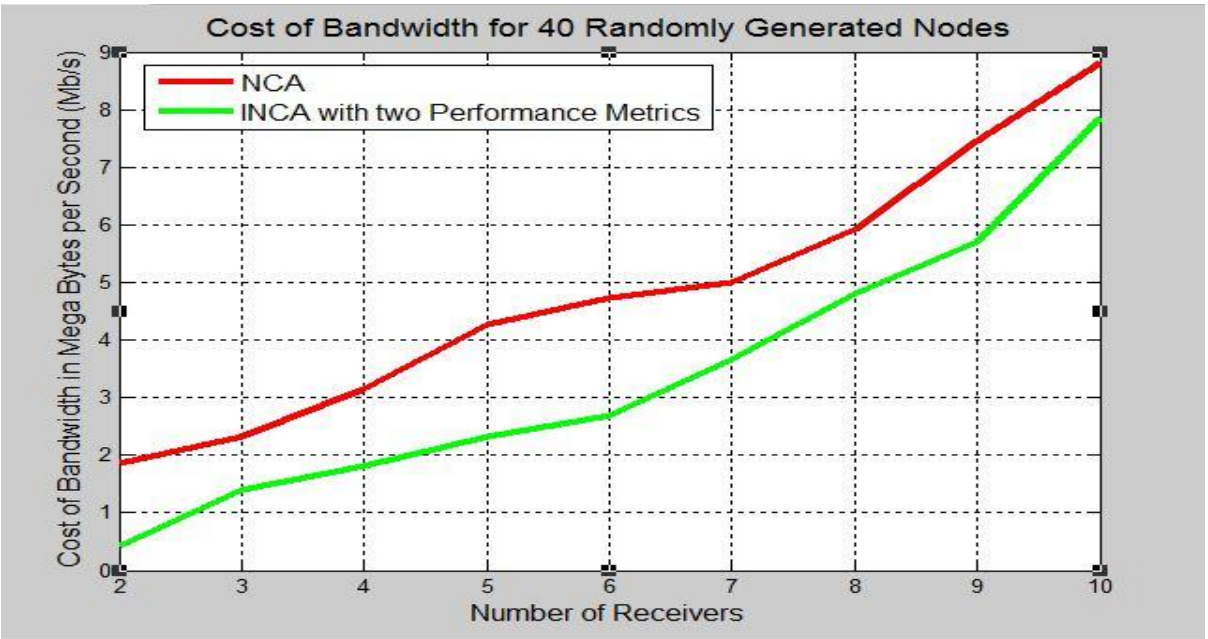


Figure 4.3: Cost of Bandwidth Multicasting 40 Randomly Generated Nodes

Figure 4.3 presents the cost of multicast for 40 randomly generated nodes to groups of Receivers. Using equation 3.2, this figure recorded the highest percentage of average cost of bandwidth reduction by 29.5% because the graph is a bit linear and it can be observed that the cost of multicast for both the two algorithm increases as the number of receivers increases.

Table 4.4: Costsof Bandwidth for 50 Randomly Generated Nodes Multicasting to Groups of Receivers.

| Number of Receivers | Cost of Bandwidth in Mega bytes per second (Mbps) | |
|---|---|---|
| | Network coding algorithm | Improved network coding algorithm |
| 2 | 0.9663 | 0.7506 |
| 3 | 2.5024 | 1.6129 |
| 4 | 3.4126 | 2.0116 |
| 5 | 4.9163 | 2.8540 |
| 6 | 5.6603 | 4.3289 |
| 7 | 6.5861 | 5.9323 |
| 8 | 7.7031 | 6.4192 |
| 9 | 8.4885 | 6.9220 |
| 10 | 8.7075 | 7.8556 |

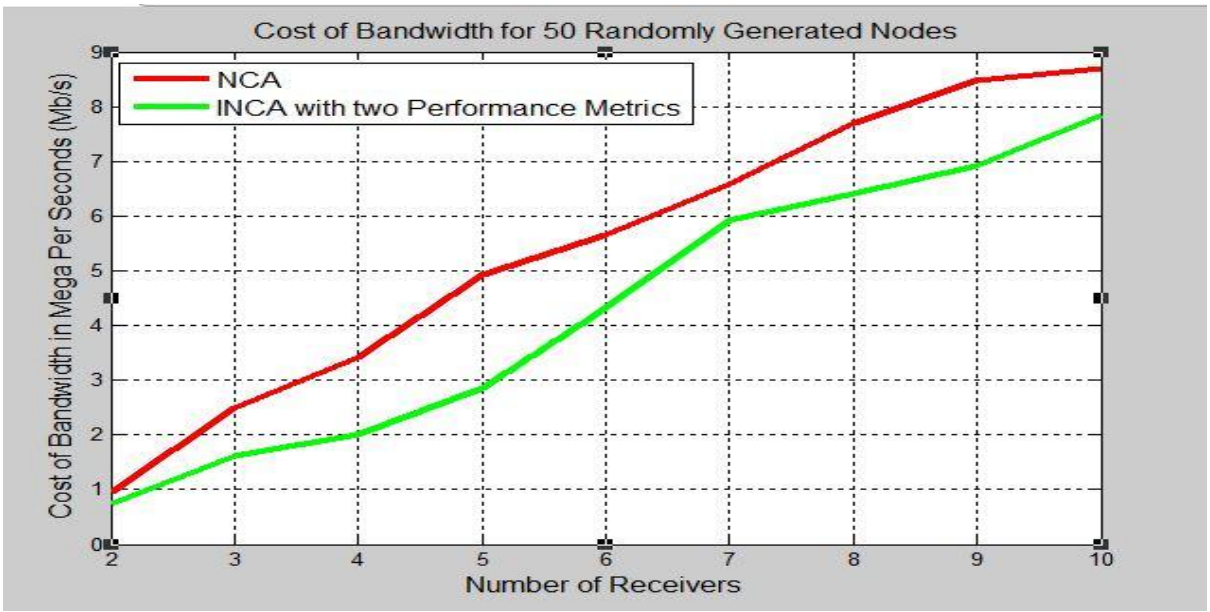



Figure 4.4 Cost of Bandwidth Multicasting 50 Randomly Generated Nodes to groups of Receivers.

Using equation 3.2, there was a reduction in the average cost of bandwidth by 20.9% and both the two algorithms and it tend to be linear.

Table 4.5: Costsof Bandwidth for 60 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of Receivers | Cost of Bandwidth in Mega Bytes per second(Mbps) | |
|---|---|---|
| | Network coding algorithm | Improved network coding algorithm |
| 2 | 2.2322 | 1.8676 |
| 3 | 2.7653 | 2.5694 |
| 4 | 1.6617 | 1.4818 |
| 5 | 5.0950 | 4.7344 |
| 6 | 6.6742 | 4.2993 |
| 7 | 5.1722 | 3.7835 |
| 8 | 6.1766 | 5.9203 |
| 9 | 8.7144 | 6.2568 |
| 10 | 8.0114 | 6.9860 |

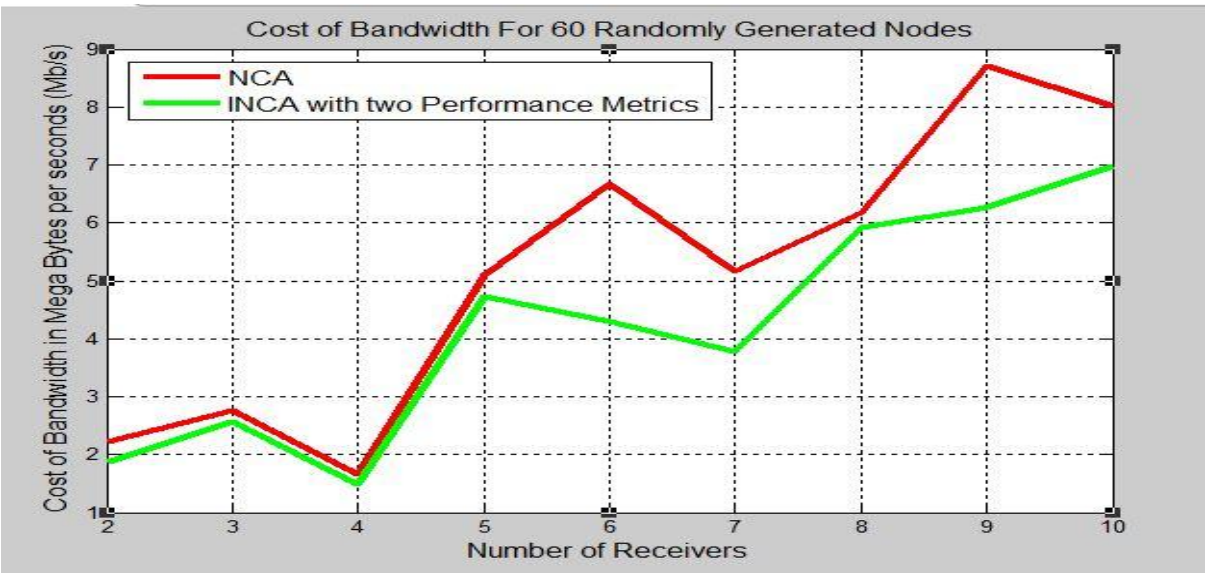



Figure 4.5: Cost of Bandwidth Multicasting 60 Randomly Generated Nodes to Groups of Receivers.

As shown in Figure 4.5 and using equation 3.2, the INCA achieved a reduction in the average cost of bandwidth by 18.5% and the two algorithms display non-linearity characteristics as the number of receivers increases. The cost of bandwidth increases and decreases when multicasting to 3, 4, 5, 6 and 7 groups of receivers because each node has different parameters and characteristics.

Table 4.6: Costsof Bandwidth for 70 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega Bytes per second (Mbps) | |
|---|---|---|
| | Network coding algorithm | Improved network coding algorithm |
| 2 | 2.1989 | 1.3733 |
| 3 | 2.4378 | 1.6463 |
| 4 | 3.9606 | 2.6533 |
| 5 | 6.5154 | 4.4382 |
| 6 | 5.8008 | 3.5805 |
| 7 | 7.5415 | 5.9904 |
| 8 | 7.3358 | 5.4878 |
| 9 | 7.5523 | 7.0702 |

| | | |
|---|---|---|
| 10 | 8.6629 | 7.3621 |

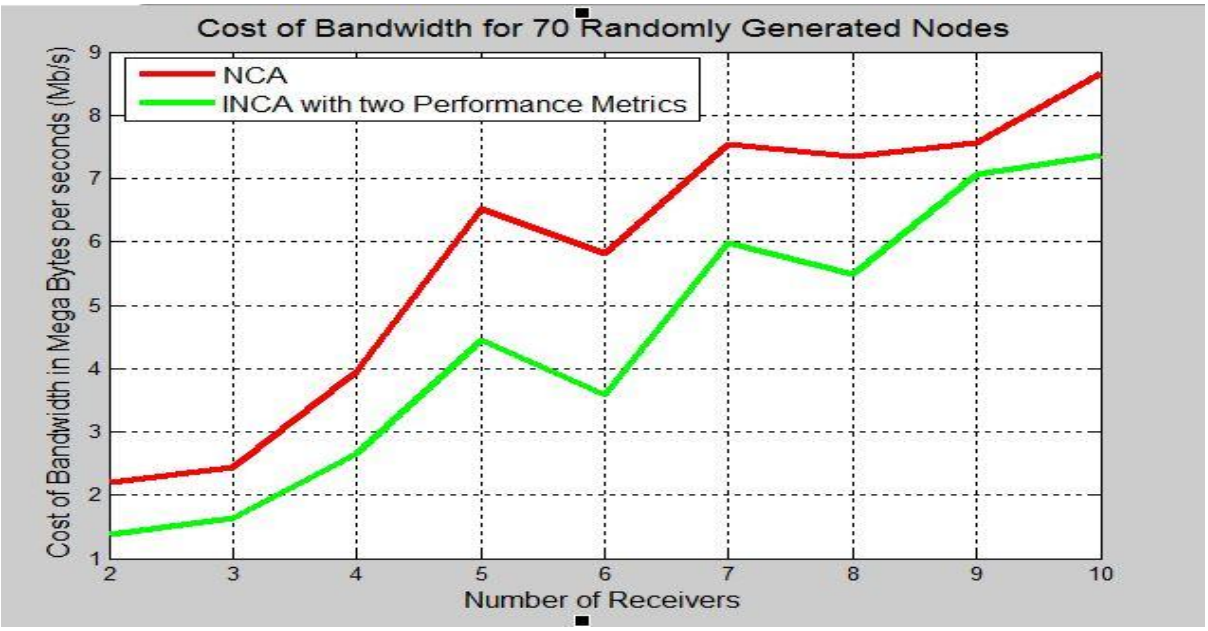



Figure 4.6: Cost of Multicasting 70 Randomly Generated Nodes to Groups of Receivers.

It can be seen that the INCA minimizes the average cost of multicast for the overall multicast session by 23.8% as compared to the NCA when computed using equation 3.2. The cost of bandwidth decreases and increases for 5, 6, 7 and 8 groups of receivers because each node has different location and distance within the multicast group.

## 4.3 PERFORMANCE EVALAUTION OF THREE ALGORITHMS

Thesummary of the results obtained for the Network Coding Algorithm, Improved Network Coding Algorithm with two parameters (packet delay and packet loss) and INCA with three parameters (packet loss, packet delay and packet congestion) are shown from Tables 4.7 to 4.12. The reduction in the average cost of bandwidth is computed using equations 3.3 and 3.4

Table 4.7: Costsof Bandwidth for 20 Randomly Generated Nodes multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega Bytes per seconds (Mbps) | | |
|---|---|---|---|
| | Network coding Algorithm | INCA with two parameters | INCA with three parameters |
| 2 | 1.0710 | 0.3405 | 0.1571 |
| 3 | 4.0521 | 1.5047 | 1.4453 |
| 4 | 3.6518 | 1.6769 | 1.5297 |
| 5 | 3.9777 | 2.2611 | 1.2306 |
| 6 | 4.0871 | 3.2449 | 2.4076 |
| 7 | 5.9811 | 4.9350 | 4.3894 |
| 8 | 7.6331 | 5.5823 | 5.4192 |
| 9 | 8.3482 | 6.2030 | 5.9324 |
| 10 | 8.5710 | 8.4356 | 7.2994 |

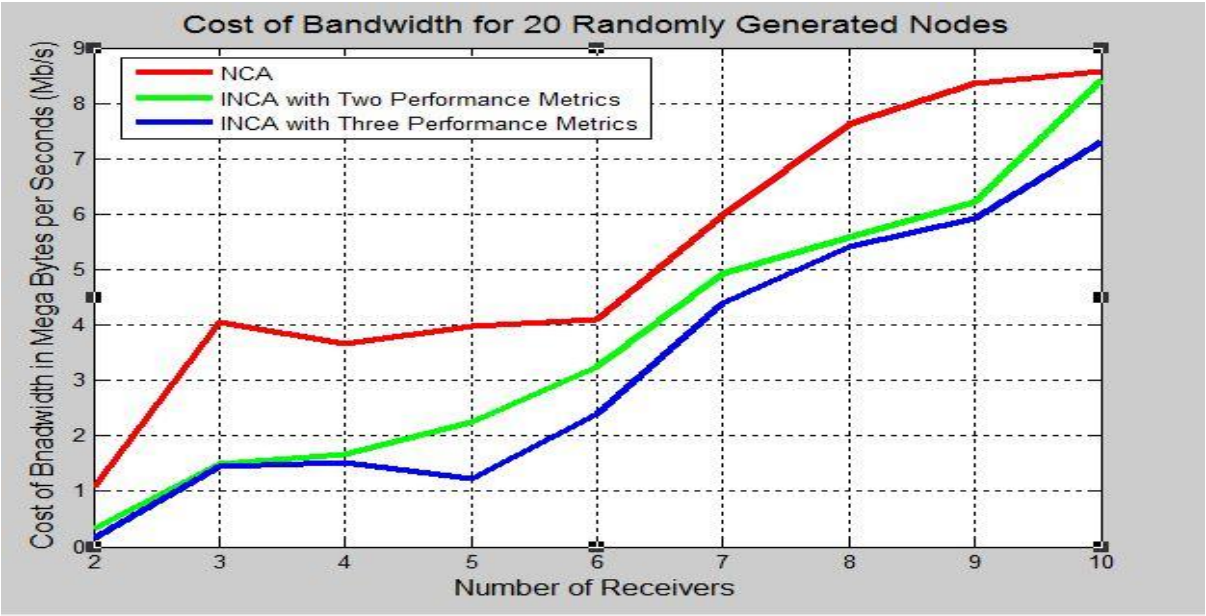



Figure 4.7: Cost of Bandwidth during Multicast for 20 Randomly Generated Nodes.

Figure 4.7 shows the cost of multicast for 20 randomly generated nodes to groups of receivers. The average cost of bandwidth decreases by 12.79% for 20 randomly generated nodes when three parameters are considered compared to considering two performance parameters.Similarly, the INCA with three performance parameters

achieved a reduction in the average cost of bandwidth by 37.07% compared with NCA.

Table 4.8: Costs of Bandwidth for 30 randomly Generated nodes multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega Bytes per seconds (Mbps) | | |
|---|---|---|---|
| | Network coding Algorithm | INCA with two Parameters | INCA with three Parameters |
| 2 | 3.6390 | 1.7120 | 0.3110 |
| 3 | 4.7158 | 1.9561 | 1.9102 |
| 4 | 3.5701 | 2.4960 | 2.1957 |
| 5 | 6.9022 | 3.8566 | 2.9439 |
| 6 | 4.6210 | 2.8661 | 2.8471 |
| 7 | 7.2319 | 4.9538 | 4.3059 |
| 8 | 6.7638 | 6.3597 | 5.9463 |
| 9 | 7.3621 | 6.6312 | 6.2401 |
| 10 | 8.9467 | 7.9821 | 6.8347 |

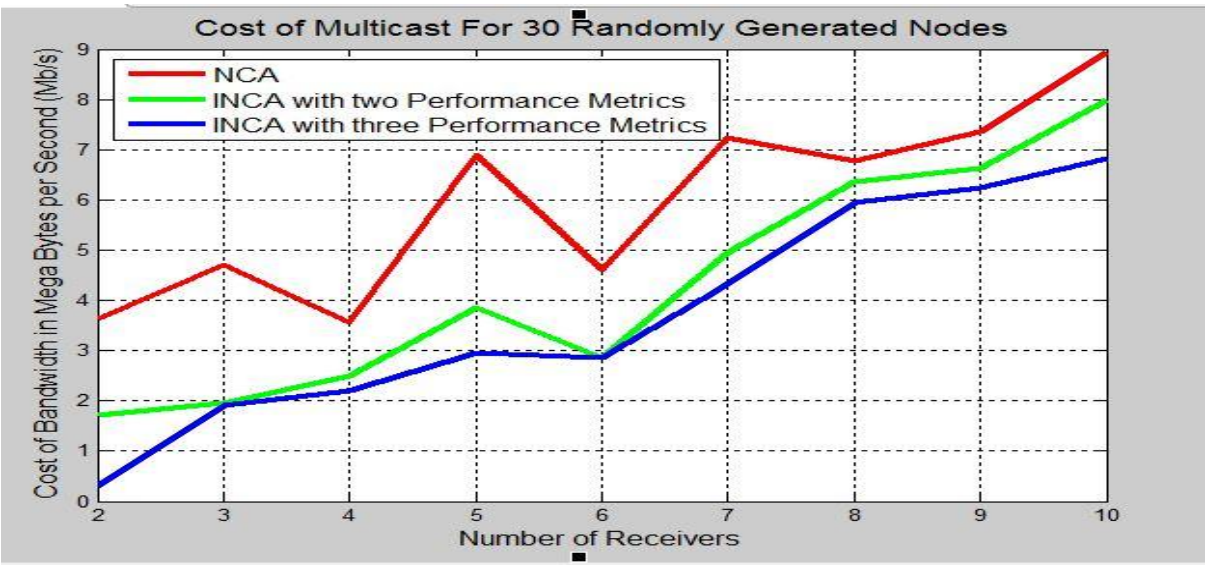



Figure 4.8: Cost of Bandwidth during Multicast for 30 Randomly Generated Nodes

Figure 4.8 shows the cost of multicast for 30 randomly generated nodes to groups of receivers. The average cost of bandwidth reduces by 13.79% for 30 randomly generated nodes when three parameters are considered when compared to only two

parameters.Similarly, the INCA with three performance parameters achieved a reduction in the average cost of bandwidth by 37.6% as compared with NCA.

Table 4.9: Costs of Bandwidth for 40 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth Multicast in Mega Bytes per second (Mbps) | | |
|---|---|---|---|
| | Network coding Algorithm | INCA with two Parameters | INCA with three Parameters |
| 2 | 1.8709 | 0.4333 | 0.1394 |
| 3 | 2.3336 | 1.3993 | 0.9004 |
| 4 | 3.1568 | 1.8031 | 1.7852 |
| 5 | 4.2686 | 2.3311 | 2.1832 |
| 6 | 4.7431 | 2.6694 | 1.8220 |
| 7 | 4.9952 | 3.6994 | 2.6633 |
| 8 | 5.9287 | 4.8068 | 4.5591 |
| 9 | 7.4452 | 5.7138 | 5.4887 |
| 10 | 8.8209 | 7.8734 | 5.9719 |

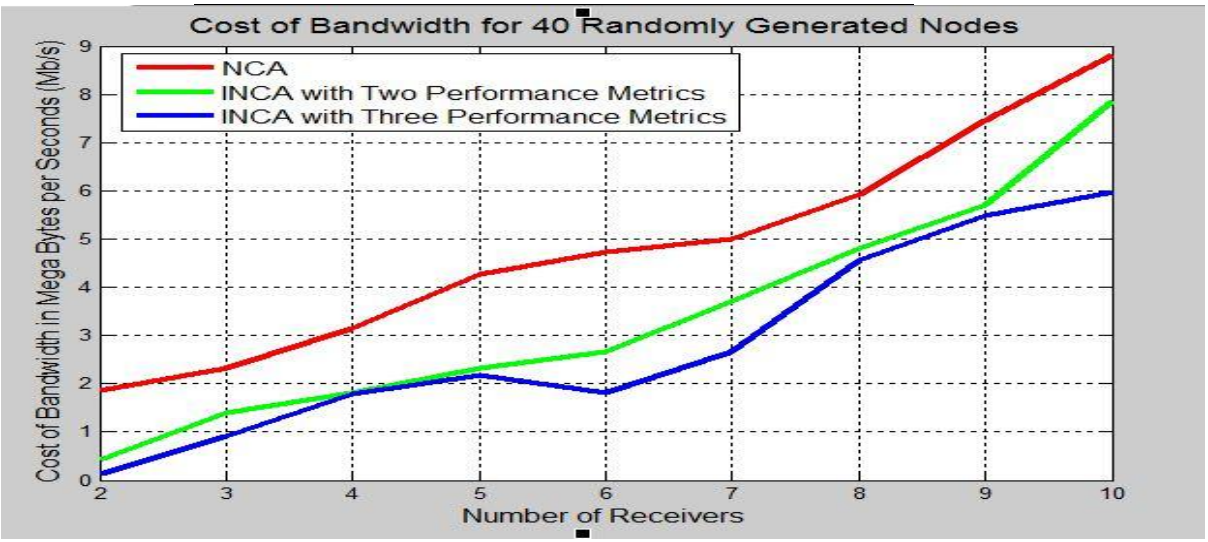



Figure 4.9: Cost of Bandwidth during Multicast for 40 Randomly Generated Nodes

Figure 4.9 shows the cost of multicast for 40 randomly generated nodes to groups of receivers. The average cost of bandwidth used during multicast when three parameters are considered decreases by 16.6% compared to the cost when two performance parameters are considered.Likewise, the INCA with three performance parameters achieved a reduction in the average cost of bandwidth by 41.15% as compared with NCA.

Table 4.10: Costs of Bandwidth for 50 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega Bytes per second (Mbps) | | |
|---|---|---|---|
| | Network coding algorithm | INCA with two parameters | INCA with three parameters |
| 2 | 0.9663 | 0.7506 | 0.2555 |
| 3 | 2.5024 | 1.6129 | 1.0598 |
| 4 | 3.4126 | 2.0116 | 1.6066 |
| 5 | 4.9163 | 2.8540 | 2.8184 |
| 6 | 5.6603 | 4.3289 | 4.3147 |
| 7 | 6.5861 | 5.9323 | 5.0145 |
| 8 | 7.7031 | 6.4192 | 6.2312 |
| 9 | 8.4885 | 6.9220 | 6.0914 |
| 10 | 8.7075 | 7.8556 | 7.0058 |

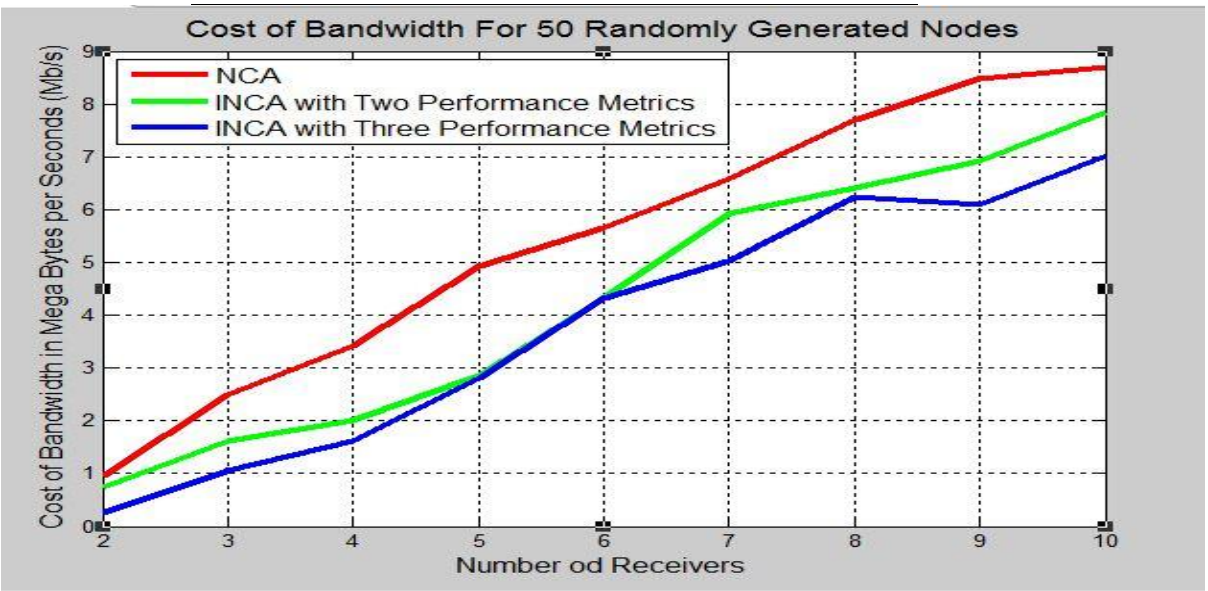



Figure 4.10: Cost of Bandwidth during Multicast for 50 Randomly Generated Nodes

Figure 4.10 shows the cost of multicast for 50 randomly generated nodes to groups of receivers. The average cost of bandwidth during Multicast when three parameters are considered decreased by 11.1% compared to a situation where only two performance metrics are considered. Equally, the INCA with three performance parameters

achieved a reduction in the average cost of bandwidth by 29% as compared with NCA.

Table 4.11: Costs of Bandwidth for 60 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega bytes per seconds (Mbps) | | |
|---|---|---|---|
| | Network coding algorithm | INCA with two parameters | INCA with three parameters |
| 2 | 2.2322 | 1.8676 | 1.0772 |
| 3 | 2.7653 | 2.5694 | 1.1341 |
| 4 | 1.6617 | 1.4818 | 1.2087 |
| 5 | 5.0950 | 4.7344 | 4.2347 |
| 6 | 6.6742 | 4.2993 | 4.1223 |
| 7 | 5.1722 | 3.7835 | 3.1230 |
| 8 | 6.1766 | 5.9203 | 5.8159 |
| 9 | 8.7144 | 6.2568 | 5.3186 |
| 10 | 8.0111 | 6.9860 | 6.5559 |

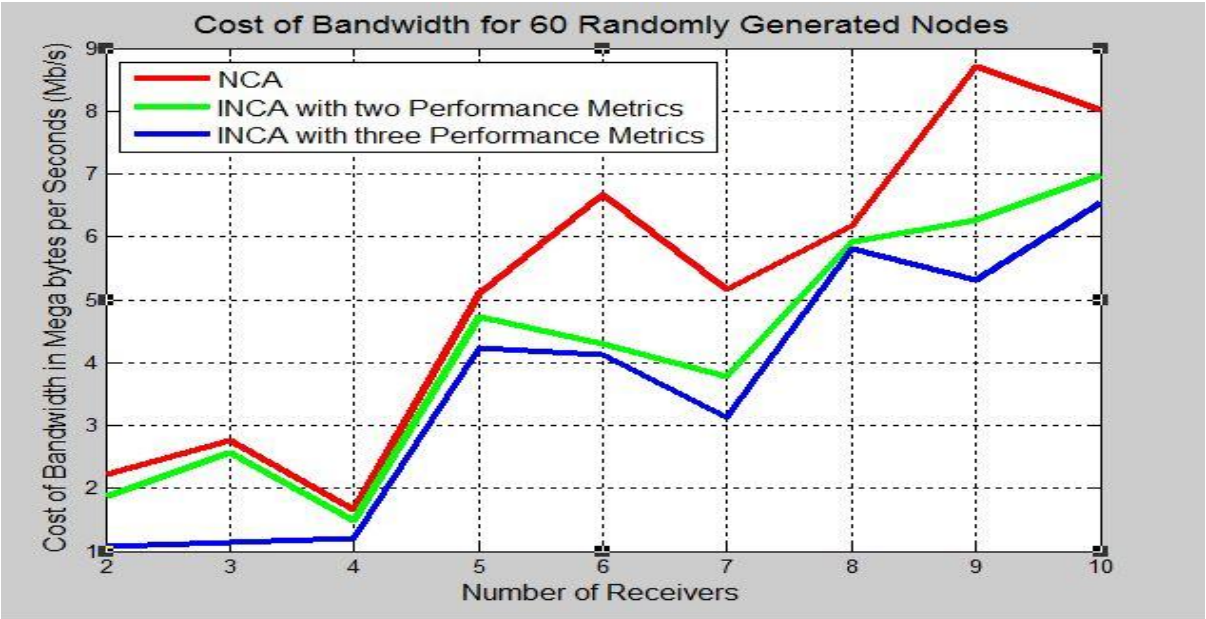



Figure 4.11: Cost of Bandwidth during Multicast for 60 Randomly Generated Nodes

Figure 4.11 shows the cost of Bandwidth during multicast for 60 randomly generated nodes to groups of receivers. The average cost of bandwidth when three performance metrics were considered for the entire session reduces by 16.28% as compared to an event when only two parameters are considered collectively. In the same way, the INCA with three performance parameters achieved a reduction in the average cost of bandwidth by 30% as compared with NCA.

Table 4.12: Costs of Bandwidth for 70 Randomly Generated Nodes Multicasting to Groups of Receivers

| Number of receivers | Cost of Bandwidth in Mega Bytes per second (Mbps) | | |
|---|---|---|---|
| | Network coding algorithm | INCA with two parameters | INCA with two parameters |

| | | | |
|---|---|---|---|
| 2 | 2.1989 | 1.3733 | 0.6202 |
| 3 | 2.4378 | 1.6463 | 1.1063 |
| 4 | 3.9606 | 2.6533 | 2.3572 |
| 5 | 5.5154 | 4.4382 | 2.8546 |
| 6 | 5.8008 | 3.5805 | 2.7041 |
| 7 | 7.5415 | 5.9904 | 5.8865 |
| 8 | 7.3358 | 5.4878 | 4.9787 |
| 9 | 7.5523 | 7.0702 | 5.9634 |
| 10 | 8.6629 | 7.3621 | 7.0899 |

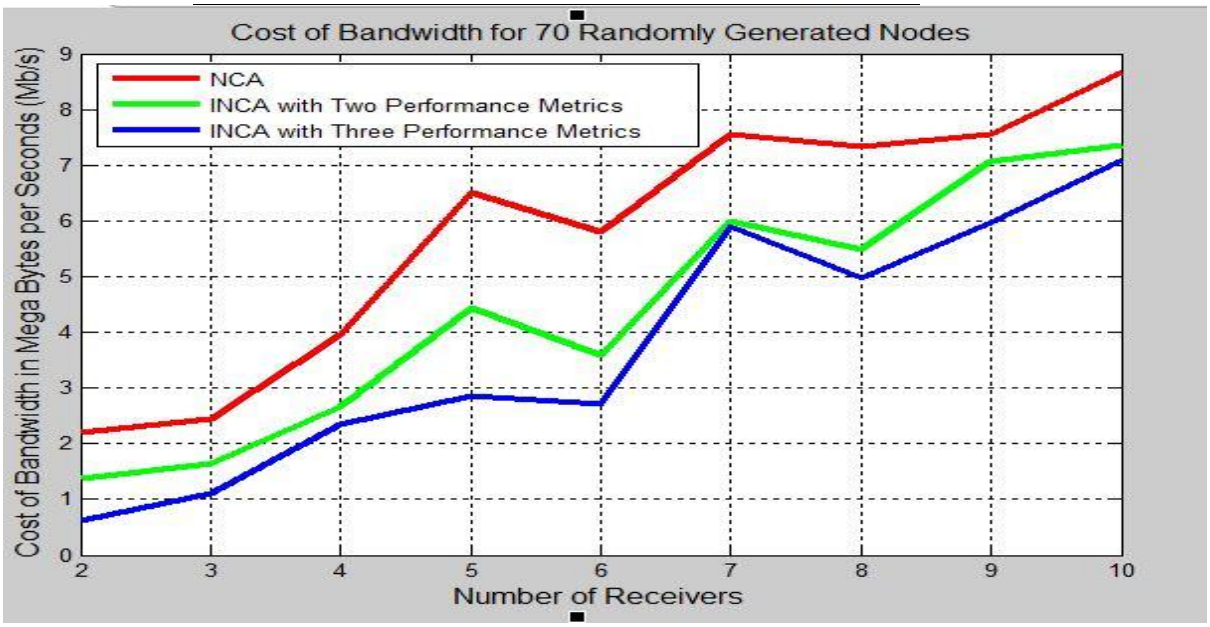



Figure 4.12: Cost of Bandwidth during Multicast for 70 Randomly Generated Nodes

From Figure 4.12, the average cost of bandwidth decreases by 15.1% when three parameters are considered compared to a situation when only two performance parameters are also considered. Also, the INCA with three performance parameters achieved a reduction in the average cost of bandwidth by 35.3% as compared with NCA.

# CHAPTER FIVE

# CONCLUSION AND RECOMMENDATION

## 5.1 INTRODUCTION

This chapter presents the conclusion and shortcomings encountered during the period of this research work as well as the direction for future research has been recommended.

## 5.2 CONCLUSION

The performance of Network Coding Algorithm (NCA) and INCA with two and three key performance metrics aimed at interpreting Shannon Hartley channel capacity theorem were examined. The average cost of bandwidth during multicast for all the numbers of randomly generated nodes decreases as more performance metrics are applied to the NCA. This research is mainly focused on investigating the problem of bandwidth utilization in accordance with metrics which affect it. Considering packet loss and packet delaymetrics collectively lead to the reduction in the average cost of bandwidth by 27%, 28.7%, 29.5%, 20.9%, 18.5% and 23.8% for all the numbers of randomly generated nodes as compared to the benchmark NCA of delay metric only. Similarly, considering packet loss, packet delay and packet rejection collectively also achieved a reduction in the average cost of bandwidth used during multicastby 37.1%, 37.6%, 41.2, 29.%, 30% and 35.3% for 20, 30, 40, 50, 60, and 70 randomly generated nodes as compared to the NCA delay metric only. More so, there was a reduction in the average cost of bandwidth when three parameters are considered by 12.7%, 13.6%, 16.6%, 11.1%, 16.3% and 15.1% as compared to a situation when two performance metrics are considered. It is evident that as more performance parameters are considered, the average cost of bandwidth decreases. It can be observed that the cost of multicast for 40 randomly generated nodes recorded the highest percentage of improvement because the cost of bandwidth tends to be linear and increases as the number of receivers equally increases.However, according to Shannon Hartley channel capacity theorem, the bandwidth increases as the noise within the channel

also increases.The cost of bandwidth presented by NCA of one metric and the INCA with two performance metrics are higher by the percentages of reduction recorded. This indicates that the cost of bandwidth in the channel is under estimated when less number of metricsis considered.The use of coded packets network has resulted in proffering many solutions to problems during multicast because of its reliability, security and resilience to node compromise and attacks.

## 5.3 SIGNIFICANT CONTRIBUTIONS

Many researchers have worked on multicast algorithms aimed at minimizing the cost of bandwidth during multicast over wireless networks. Many researches have also been carried out on different areas of the Multimedia applications using different algorithms. The significant contributions of this research work are as follows:

i. The study established that the more the numbers of key performance metrics are considered, the better the true picture of the channel is represented and the accurate cost of bandwidth is achieved.

ii. There was a decrease in the average cost of bandwidth used during multicast when two performance metrics (packet delay and packet loss) are considered by 37.1%, 37.6%, 41.2%, 29%, 30% and 35.3% for all the number of randomly generated nodes within the network as compared to the benchmark algorithm.

iii. Similarly, the average cost of bandwidth decreases further when three performance metrics (packet delay, packet loss and packet rejection) are considered by 12.7%, 13.6%, 16.6%, 11.1%, 16.3% and 15.1% for all the numbers of randomly generated nodes within the network when compared to a situation when two performance metrics are considered.

iv. The study also established that the cost of bandwidth within the channel was under estimated as more performance metrics are considered in the channel.

## 5.4 LIMITATIONS

The limitations of this research work are itemized as follows:

i. All the performance metrics that affect multicast were not considered to further investigate their impact on the algorithm.

ii. The research was implemented using simulation rather than in real time situation.

iii. All performance metrics affecting the channel and bandwidth were not exhaustively considered, so the limit of the impact on the cost of bandwidth was not reached.

## 5.5 RECOMMENDATIONS FOR FURTHER WORK

The following possible areas are recommended for future work:

i. The implementation of the research in real time should be reflected by considering two or three performance parameters.

ii. Packet congestion and interference should be considered in addition to all the performance metrics considered in this research.

iii. Dynamic multicast should also be considered with fewperformance metrics mentioned.

## REFERENCES

Ackermann, H., Ralglin, H., & Valcking, B. (2008). On the impact of combinatorial structure on congestion games. *Journal of the ACM (JACM), 55*(6), 25.

Afolabi, R. O., Dadlani, A., & Kim, K. (2013). Multicast scheduling and resource allocation algorithms for OFDMA-based systems: A survey. *Communications Surveys & Tutorials, IEEE, 15*(1), 240-254.

Ahlswede, Cai, N., Li, S. Y. R., and , & Yeung, R. W. (2000). Network information flow. Information Theory. *Theory, IEEE Transactions on, 46*(4), 1204-1216.

Ajibesin, A. A., Ventura, N., Chan, H. A., Murgu, A., & Egunsola, O. K. (2012). *Performance of Multicast Algorithms Over Coded Packet Wireless Networks.* Paper presented at the Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on.

Ajibesin, A. A., Ventura, N., Murgu, A., & Chan, H. A. (2014). Data envelopment analysis with slacks model for energy efficient multicast over coded packet wireless networks.

Ajibesin, A. A., Wajiga, G. M., Odekunle, M. R., & Egunsola, O. K. (2013). Energy-efficient Multicast in wired and wireless Networks: Analysis and performance Measures. *Computer science and artificail intellegent laboratory, 65*, 23-29.

Anna, L., Roberto Tamassia, & Triandopoulos, N. (2004). Multicast Authentication in Fully Adversarial Networks. *A Paper presented in 2004 IEEE Symposium on Security and Privacy*, 1-29.

Annapurna, D., Tejas, N., Raja, K. B., & Venugopal, K. R. (2014). Energy Efficient Multicast Algorithm for an Ad Hoc Network.

Anshelevich, E., Dasgupta, A., Kleinberg, J., Tardos, E., Wexler, T., & Roughgarden, T. (2008). The price of stability for network design with fair cost allocation. *SIAM Journal on Computing, 38*(4), 1602-1623.

Avni, G., Kupferman, O., & Tamir, T. (2014). *Network-Formation Games with Regular Objectives.* Paper presented at the Foundations of Software Science and Computation Structures.

Banchs, A., De la Oliva, A., Eznarriaga, L., Kowalski, D., & Serrano, P. (2014). Performance Analysis and Algorithm Selection for Reliable Multicast in IEEE 802.11 aa Wireless LAN.

Bang, Y. W., & Chao, K.-M. (2009). Introduction to short path tree.

Bassoli, Hugo Marques, Jonathan Rodriguez, Kenneth W. Shum, & Tafazolli, R. (2013). Network Coding Theory: A Survey. *IEEE Communications Surveys & Tutorials, 15*(4).

Bhadra, S., Shakkottai, S., & Gupta, P. (2006). Min-cost selfish multicast with network coding. *Information Theory, IEEE Transactions on, 52*(11), 5077-5087.

Bley, A. (2011). An integer programming algorithm for routing optimization in IP networks. *Algorithmica, 60*(1), 21-45.

Bolot, J.-C. (1993). Characterizing end-to-end packet delay and loss in the internet. *Journal of High Speed Networks, 2*(3), 305-323.

Bolot, J.-C. (1993). End-to-end packet delay and loss behavior in the Internet. *ACM SIGCOMM Computer Communication Review, 23*(4), 289-298.

Bosch, R., & Trick, M. (2004). Integer programming chapter 3. (70-94).

Boyd, S. P., & Vandenberghe, L. (2004). Convex optimization.

Byrka, J., & Grandoni, F. (2010). An improved LP-Base approximation for steiner tree. *proceedings of the 42nd ACM symposium on theory of computing*, 583-592.

Castillo, E., Conejo A.J., Pedregal, P., Garca, R., & Alguacil, N. (2002). Building and Solving Mathematical Programming Models in Engineering and Science, Pure and Applied Mathematics Series, Wiley, New York. *International Journal of scientific and Engineering research*, 14-18.

Chen, Ho, T., Chiang, M., Low, S. H., & Doyle, J. C. (2012). Congestion control for multicast flows with network coding. *Information Theory, IEEE Transactions on, 58*(9), 5908-5921.

Chen, W., Letaief, K. B., & Cao, Z. (2007). *Opportunistic network coding for wireless networks.* Paper presented at the Communications, 2007. ICC'07. IEEE International Conference on.

Choi, B.-Y., Moon, S., Zhang, Z.-L., Papagiannaki, K., & Diot, C. (2007). Analysis of point-to-point packet delay in an operational network. *Computer Networks, 51*(13), 3812-3827.

Chopra, K., & Mishra, A. (2012). Packet Rejection Based on Packet Rank for Congestion Control. *management, 1*(3).

Cui, T., Chen, L., & Ho, T. (2008). *Energy efficient opportunistic network coding for wireless networks.* Paper presented at the INFOCOM 2008. The 27th Conference on Computer Communications. IEEE.

Dau, S. H., Dong, Z., Yuen, C., & Chan, T. H. (2013). Delay Minimization in Varying-Bandwidth Direct Multicast with Side Information. *arXiv preprint arXiv:1301.6433*.

Debbra, W. (2003). *OSI Reference Model for Telecommunications*: McGraw- Hill companies.

Deng, S. (1995). *Traffic characteristics of packet voice.* Paper presented at the Communications, 1995. ICC'95 Seattle,'Gateway to Globalization', 1995 IEEE International Conference on.

Dimakis, A. G., Godfrey, P. B., Wu, Y., Wainwright, M. J., & Ramchandran, K. (2010). Network coding for distributed storage systems. *Information Theory, IEEE Transactions on, 56*(9), 4539-4551.

Dimitri Bertsekas, & Gallager, R. (2001). Data Networks. In S. Edition (Ed.), (pp. 495-506): Prentice Hall. Englewood Cliffs, New Jersey 07632.

Dougherty, R., Freiling, E., & Zeger, K. (2014). Characteristic-Dependent Linear Rank Inequalities with Applications to Network Coding. *arXiv preprint arXiv:1401.2507*.

Douik, A., Sorour, S., Al-Naffouri, T. Y., Yang, H.-C., & Alouini, M.-S. (2014). Delay Reduction in Multi-Hop Device-to-Device Communication using Network Coding. *arXiv preprint arXiv:1409.8460*.

Dutta, B., & Mishra, D. (2012). Minimum cost arborescences. *Games and Economic Behavior, 74*(1), 120-143.

El Rouayheb, S. Y., Georghiades, C. N., & Sprintson, A. (2006). *Network coding in minimal multicast networks.* Paper presented at the Information Theory Workshop, 2006. ITW'06 Punta del Este. IEEE.

Fragouli, C., & Soljanin, E. (2007). *Network coding fundamentals*: Now Publishers Inc.

Frost, D., & Bryant, S. (2011). Packet Loss and Delay Measurement for MPLS Networks.

Guan, L., He, Y., & Kung, S.-Y. (2012). *Multimedia image and video processing*: CRC press.

Hamza, F. A., Romdhani, L., & Mohamed, A. (2013). Novel Network Coding-based Techniques for Multi-layer Video Delivery over Multi-hop Wireless testbed. *arXiv preprint arXiv:1304.7578*.

Han, Z.-j., Wang, R.-c., & Xiao, F. (2014). A Multicast Algorithm for Wireless Sensor Networks Based on Network Coding. *International Journal of Distributed Sensor Networks, 2014*.

Haripriya, E., & Valluvari, K. R. (2013). A Network survey and a Comparative Study of Multicast Routing Techniques in Disconnected Adhoc. *International journal of scientific research engineeering & Technology 2*(4), 237-241.

Herzog, S., Shenker, S., & Estrin, D. (1997). Sharing the "cost" of multicast trees: an axiomatic analysis. *Networking, IEEE/ACM Transactions on, 5*(6), 847-860.

Hougardy, S., Silvanus, J., & Vygen, J. (2014). Dijkstra meets Steiner: a fast exact goal-oriented Steiner tree algorithm. *arXiv preprint arXiv:1406.0492*.

Hu, X.-M., & Zhang, J. (2013). Minimum Cost Multicast Routing Using Ant Colony Optimization Algorithm. *Mathematical Problems in Engineering, 2013*.

Immorlica, N., Mahdian, M., & Mirrokni, V. S. (2008). Limitations of cross-monotonic cost-sharing schemes. *ACM Transactions on Algorithms (TALG), 4*(2), 24.

Jain, K., & Mahdian, M. (2007). Cost sharing. *Algorithmic game theory*, 385-410.

Kandari, S., & Pandey, M. K. (2014). QoS Multicasting in MANET.

Karaoglu, B., & Heinzelman, W. (2010). *Multicasting vs. Broadcasting: What are the Trade-offs?* Paper presented at the Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE.

Karthikeyan, E., & Shankar, R. (2014). VoIP Packet Delay Techniques: A Survey. *Global Journal of Computer Science and Technology, 14*(3).

Lagercrantz, H. (2014). Multicasting in an online gaming system: Google Patents.

Lertpratchya, D., Blough, D. M., & Riley, G. F. (2014). *Interference-aware multicast for wireless multihop networks.* Paper presented at the Wireless Communications and Networking Conference (WCNC), 2014 IEEE.

Li, S. Y., Yeung, R. W., & Cai, N. (2003). Linear network coding. *Information Theory, IEEE Transactions on, 49*(2), 371-381.

Li, Z., & Chu, X. (2012). On achieving group-strategyproof multicast. *Parallel and Distributed Systems, IEEE Transactions on, 23*(5), 913-923.

Liu, J., Jiang, X., Nishiyama, H., & Kato, N. (2012). *Multicast capacity, delay and delay jitter in intermittently connected mobile networks.* Paper presented at the INFOCOM, 2012 Proceedings IEEE.

Lopes, P. R. S., Silva, Y. C. B., & Cavalcanti, F. R. P. (2010). *Efficient wireless multicast retransmission techniques based on multiple coded packets.* Paper presented at the Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd.

Lun, D. S., Ratnakar, N., Macdard, M., Koetter, R., Karger, D. R., Ho, T., *et al.* (2006). Minimum-cost multicast over coded packet networks. *Information Theory, IEEE Transactions on, 52*(6), 2608-2623.

Luo, Z.-Q., & Yu, W. (2006). An introduction to convex optimization for communications and signal processing. *Selected Areas in Communications, IEEE Journal on, 24*(8), 1426-1438.

Ma, & Denko, M. K. (2008). *Wireless quality of service: techniques, standards, and applications*: CRC Press.

Ma, L., Lin, Z., Zhang, Z., Mao, G., & Vucetic, B. (2013). *Improving reliability in lossy wireless networks using network coding.* Paper presented at the Communications Workshops (ICC), 2013 IEEE International Conference on.

Maher, H., Raaele N, & Hubaux, J.-P. (1998). Fresh Packet First Scheduling for Voice Trafic in Congested Networks

Mani, D. M. (2014). Secure Multicasting for Wireless Sensor Networks. *IJCSNS, 14*(11), 70.

Minhas, T. N., Lagunas, O. G., Arlos, P., & Fiedler, M. (2012). *Mobile video sensitivity to packet loss and packet delay variation in terms of QoE.* Paper presented at the Packet Video Workshop (PV), 2012 19th International.

Montessoro, P. L. (2013). Efficient management and packets forwarding for multimedia flows. *Journal of network and systems management, 21*(2), 194-217.

Mwela, J. S. (2010). *Impact of Packet Loss on the Quality of Video Stream Transmission.* Blekinge Institute of Technology.

Nemirovski, A. (2006). Advance in convex optimization: conic programming. *in proceedings on the international congress for Mathematician: Madrid, ,* 413-444.

Nguyen, U. T., & Xu, J. (2007). Multicast routing in wireless mesh networks: Minimum cost trees or shortest path trees? *Communications Magazine, IEEE, 45*(11), 72-77.

Norde, H., Moretti, S., & Tijs, S. (2004). Minimum cost spanning tree games and population monotonic allocation schemes. *European Journal of Operational Research, 154*(1), 84-97.

Oliveira, C. A. S., & Pardalos, P. M. (2005). A survey of combinatorial optimization problems in multicast routing. *Computers & Operations Research, 32*(8), 1953-1981.

Paschos, G. S., Li, C.-p., Modiano, E., Choumas, K., & Korakis, T. (2014). *Multirate Multicast: Optimal Algorithms and Implementation.* Paper presented at the IEEE Infocom.

Perkins, C., Hodson, O., & Hardman, V. (1998). A survey of packet loss recovery techniques for streaming audio. *Network, IEEE, 12*(5), 40-48.

Qadir, J., Baig, A., Ali, A., & Shafi, Q. (2014). Multicasting in cognitive radio networks: Algorithms, techniques and protocols. *Journal of Network and Computer Applications, 45*, 44-61.

Rajkumar, K. (2014). Efficient Resource Allocation in Multicasting over Mobile Adhoc Networks. *Indian Journal of Science and Technology, 7*(S5), 71-75.

Ramamoorthy, A. (2011). Minimum cost distributed source coding over a network. *Information Theory, IEEE Transactions on, 57*(1), 461-475.

Rardin, R. L. (1998). *Optimization in operations research* (Vol. 166): Prentice Hall New Jersey.

Rashed, A. N. Z. (2012). Transmission performance evaluation of optical add drop multiplexers (OADMs) in optical telecommunication ring networks. *American Journal of Engineering and Technology Research Vol, 11*(10).

Rhee, I. (1998). *Retransmission-based error control for interactive video applications over the internet.* Paper presented at the Multimedia Computing and Systems, 1998. Proceedings. IEEE International Conference on.

Sanders, P., Egner, S., & Tolhuizen, L. (2003). *Polynomial time algorithms for network information flow.* Paper presented at the Proceedings of the fifteenth annual ACM symposium on Parallel algorithms and architectures.

Schalkamp, J. (2012). Minimum-Cost Multicast over Coded Packet Networks. *network Architecture and Services, 67*(seminar: innovative internet technology and mobile communications WS11/12), 34-41.

Shannon, C. E. (1948). A Mathematical theory of Communication. *Proc. IRE, 27*, 379-423.

Shannon, C. E. (1949). Communication in the Presence of Noise. *Proc. IRE, 37*(1), 10-21.

Shin, Y., Choi, M., Koo, J., & Choi, S. (2013). Video multicast over WLANs: Power saving and reliability perspectives. *Network, IEEE, 27*(2), 40-46.

Shrader, B., & Ephremides, A. (2012). Queueing delay analysis for multicast with random linear coding. *Information Theory, IEEE Transactions on, 58*(1), 421-429.

Singh, K., & Yadav, R. S. (2012). Multicast Congestion Control in Adversary Environment. *International Proceedings of Computer Science & Information Technology, 31*.

Skorin-Kapov, D. (2001). On cost allocation in Steiner tree networks *Steiner Trees in Industry* (pp. 353-375): Springer.

Sommers, J., Barford, P., & Duffield, N. a. R., A. (2005). Improving Accuracy in End to end packet loss Measurement

Soumen, K., & Mohammad, S. (2010). Performance of Multirate Multicast in Distributed Network. *Int'l J. of Communications, Network and System Sciences, 2010*.

Striegel, A., & Manimaran, G. (2002). A survey of QoS multicasting issues. *communication magazine, IEEE, 40*(6), 82-87.

Swapna, B. T., Eryilmaz, A., & Shroff, N. B. (2013). Throughput-delay analysis of random linear network coding for wireless broadcasting. *Information Theory, IEEE Transactions on, 59*(10), 6328-6341.

Tan, Siew, Z. W., Chin, K. Y., Lye, K. C. S., & Teo, K. T. K. (2012). Minimizing network coding Nodes in Multicast Tree construction via Genetic Algorithm. *Computer society*.

Tan, G., Ni, X., Liu, X., Qu, C., & Tang, L. (2012). Real-time multicast with network coding in mobile ad-hoc networks. *Intelligent Automation & Soft Computing, 18*(7), 783-794.

Tang, C., & McKinley, P. K. (2003). *Modeling multicast packet losses in wireless LANs.* Paper presented at the Proceedings of the 6th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems.

Tayu, S., Al-Matairi, G. T., & Ueno, S. (2005). Cost-constrained Minimum delay Multicasting *spring -Verga Berlin Heidelberg, 3*(SOFSEM), 330-339.

Vielma, J. P. (2014). Mixed Integer Programming Formulation Techniques. (Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts.).

Wen, Y.-F., & Liao, W. (2010). Minimum power multicast algorithms for wireless networks with a Lagrangian relaxation approach. *Wireless Networks, 17*(6), 1401-1421.

Wenzhong, Y., & Zhenyu, Z. (2013). A Qos Multicast Routing Protocol in WSN Network.

Wu, F., Sun, Y., Yang, Y., Srinivasan, K., & Shroff, N. B. (2014). Constant Delay and Constant Feedback Moving Window Network Coding for Wireless Multicast: Design and Asymptotic Analysis. *arXiv preprint arXiv:1404.3438*.

Xi, Y., & Yeh, E. M. (2010). Distributed algorithms for minimum cost multicast with network coding. *Networking, IEEE/ACM Transactions on, 18*(2), 379-392.

Xi, Y., & Yeh, E. M. (2011). Distributed algorithms for minimum cost multicast with network coding. *Networking, IEEE/ACM Transactions on, 18*(2), 379-392.

Xiong, C., & Li, X. (2010). *Minimum cost optimization of multicast wireless networks with network coding.* Paper presented at the Information Sciences and Systems (CISS), 2010 44th Annual Conference on.

Xiumin, W., Jin, W., & Shukui, Z. (2012). Network coded wireless coopeartive Multicast with Minimum Transmission cost. *international journal of distributed sensor networks, 5*, 1-12.

Xuan, Y., & Lea, C.-T. (2011). Network-coding multicast networks with QoS guarantees. *IEEE/ACM Transactions on Networking (TON), 19*(1), 265-274.

Yang, C., & Zhao, X. (2014). A New Delay-Constrained Multicast Routing Algorithm Based on Shared Edges. *Communications and Network, 2014*.

Yin, X., Wang, Y., Li, Z., Wang, X., Zhao, J., & Xue, X. (2014). Bounding the Advantage of Multicast Network Coding in General Network Models.

Zhang, H., Zhou, J., & Li, J. (2011). Minimising delay for video conference with network coding. *International Journal of Information and Coding Theory, 2*(1), 38-58.

Zhansong, M. (2005). Multicast Source Routing Algorithms. *Computer science and artificail intelligent laboratory, 3*, 331-339.

Zhao, C., Luo, Y., Chen, F., Zhang, J., & Wang, R. (2014). Energy Effective Congestion Control for Multicast with Network Coding in Wireless Ad Hoc Network. *Mathematical Problems in Engineering, 2014*.

Zhu, Q., Parsa, M., & Garcia-Luna-Aceves, J. J. (1995). *A source-based algorithm for delay-constrained minimum-cost multicasting.* Paper presented at the INFOCOM'95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings. IEEE.

Zongpeng, L. (2008). *Cross-monotonic multicast.* Paper presented at the INFOCOM 2008. The 27th Conference on Computer Communications. IEEE.

## APPENDIX A

**(Code)**

This file reads as input the output of number of randomly generated nodes from MIPA and keeps all the information about the nodes and its distance in an array.

```
void filetest(char *filename, double **data_ptr, int *num_edges)
{

 // this Variables are used for extracting information about each link.

 int out, in, DEBUG =0;   // out =represent  starting node of each link; in = represent ending node of
each link;
 double lat;                              // lat = link distance from the starting node to the ending node;
 int line_cnt=0, i = 0;

 FILE *file_in;              // this represent the internal variable assigned to input file

 // this code is use for testing the I/O stream from input file that is the test file generated by MIP;

 char junk1[50], name[50];
 int junk;

 file_in = fopen(filename, "r");
 if(file_in ==NULL){puts("zero"); return;};

 while(1)
  {
   fscanf(file_in, "%s%d%s%d%d", junk1, &junk, junk1, &junk, &junk);
   if(feof(file_in) != 0) break;
   line_cnt++;
  };

 fclose(file_in);

 // this command Check for the total number of edges and passes the address of data to be stored;

 double *M = (double *)malloc(line_cnt*3*sizeof(double));

 file_in = fopen(filename, "r");
 if(file_in ==NULL){puts("There is no such file"); return;};

 while(1)
  {
   fscanf(file_in, "%s", name);      // read starting node for this link and its name;

   if (feof(file_in) != 0)
        break;                                  // reached end of file - break out of while loop;
   fscanf(file_in, "%d", &out);      // read the index of the starting node for this link;
   fscanf(file_in, "%s", name);     // read the name of the ending node for this link;
   fscanf(file_in, "%d", &in);       // read the value of packet delay as formulated;
   fscanf(file_in, "%lf",  &lat);   // read the distance for this link;
    fscanf(file_in, "%lf",  &in);   // read the value of packet loss  as formulated;
   fscanf(file_in, "%s",  &in);    // read the value of packet rejection as formulated;

   M[i] = out;                       // save the index of the starting node in array M;
   M[i+1] = in;                      // save the index of the ending node in array M;
   M[i+2] = lat;                     // save also the distance of this link in array M;

   i=i+4;                                  // increase the array pointer by 4 (to store values for
                                     // the next link)
  };

 if(DEBUG){printf("E = %d, M = %f \n",i/4, M[0]);};
```

```
fclose(file_in);

*data_ptr = M;
*num_edges = (int)(i/4);
```

## APPENDIX B
## Formation of A_Z Matrix

This file takes the network data in an array that is output from test file. The array contains all pertinent information about the nodes and the distance. This file extracts this information from the array and

forms an A_Z matrix based on the wireless constraints formulated as mixed integer programming described in this research work.

```
/* the A_Z matric is formulated by the following method */

void formulate(double* data_ptr, int e, int t,int d, int pl, double R, double* A, int* A_z, int* A_x,
double **Bp, int *C, int *v, int *t1, int *count, int wireless)
{

  /* Constraint Matrix:
     t: number of sinks,
     e: number of edges;
     v_cnt: number of nodes;
d: packet delay, (400)
     pl: packet loss; (4)
     pr: packet rejection (8)

     |A_z| |X_z| = |b|
     |A_x| |X_x|          */

  /* Defining A_z Matrix (e(t+1) by et)*pl
     This represents the additional variable constraint (packet loss) considered in this research:
     z_e - x_e^t != 0*/


  //int *A_z = (int *) calloc((2e+1)*e*t*3*400, sizeof(int)); // e*(e+2)*t*4*8 elements; 3 entries for
each

  int inx = 0, cnt=0;

   (wireless)
  {
   printf("\nPerforming wireless optimization: \n");  the algorithm takes care of packet loss from zero
(0) to four (4) and the value of packet delay (400) for maximum flow of packets during multicast.

   for(j=0;j<t*e;j=j+t)
    {
            Int_d [400];         // collect all indices with cost >= A[cnt]

            for(d=0; d<e; d++)
       {

              if(A[k] >= A[cnt]) {
                 temp[d] = 400;
          }
          else temp[d] = 0;
       }             // collected in temp


            for(i=0; i<t; i++)
     {
             for(d=0; d<e; d++)
             {
               if(temp[d] == 1 && list[2*d] == list[2*cnt])  // edge has cost >= and is emanating from the
same node
                  {
             A_z[inx] = j+i;   // row j+i
                    A_z[inx+1] = d;   // column k, representing link z_(d+1)
             A_z[inx+2] = 1;   // set entry = 1
                //  printf("z(%d, %d) ", A_z[inx], A_z[inx+1]);
```

```
          A_z[inx+3] = j+i; // row j+i
                A_z[inx+4] = e+e*t+d*t+((j+i)%t); // locate the appropriate x_d^t
          A_z[inx+5] = -1;  // set entry = -1
             //  printf("x(%d, %d) ", A_z[inx+3], A_z[inx+4]);

                      inx += 6;
            counter += 2;
              }
       }

              A_z[inx] = j+i;       // slack variable
              A_z[inx+1] = j+i+e;
              A_z[inx+2] = -1;
           //   printf("s(%d, %d)\n", A_z[inx], A_z[inx+1]);

           inx=inx+3;
     counter++;
    };              // for(i=0;...);
    cnt++;
  };                // for(j=0;...);
 }                  // (wireless)

/* Defining A_x Matrix:
          v_cnt   : enter the number of nodes within the multicast group
               e    : enter the of edges within the multicast group
               nodes[i]: list of nodes in an increasing order in the group
               list[i]: list of edges (nodeA->nodeB)
               size    : v_cnt by e

   Suppose *NodeA->NodeB edge is ith edge in the slist,
          *Node A is pth in the nodes of the multicast group,
          *Node B is qth in the nodes. Then,

   pth row ith col = 1 (outgoing)
   qth row ith col = -1 (incoming)
   This matrix is reformed suitably to represent t flows of packets:
*/

//int *A_x = (int *)calloc((2*e*t)*3, sizeof(int));

inx = 0;

for(i=0; i<e*2; i=i+2)
  {
    for(j=0;j<v_cnt;j++)
          {
            if(list[i] == nodes[j]) {p = j;}
            if(list[i+1] == nodes[j]){q = j;}
          }

    for(k=0;k<t;k++)
          {
            A_x[inx] = (e*t)+t*p+k;
            A_x[inx+1] = (e*(t+1))+ t*(i/2)+k;
            A_x[inx+2] = 1;

            A_x[inx+3] = (e*t)+t*q+k;
            A_x[inx+4] = (e*(t+1))+t*(i/2)+k;
            A_x[inx+5] = -1;
```

```
            inx= inx+6;
        counter += 2;
            };
  };

  for(i=0; i<2*e; i=i+2)
  {
   printf("%d %d \n", list[i], list[i+1]);
  };
//Testing
if (oDEBUG)
  {
    printf("%s\n", "List");
    for(i=0;i<2*e;i=i+2)
          {printf("%d %d \n", slist[i], slist[i+1]);};};

if (oDEBUG)
  {
    printf("%s\n", "Node List:");
    for (i=0;i<v_cnt;i++)
          {printf("%d: %d\n", i, nodes[i]);};
    printf("\n");
  };

if(oDEBUG)
  {for(i=0;i<2*e*t*3/6;i++)
    {for(j=0;j<6;j++)
          {printf("%d ", A_x[i*6+j]);}
    printf("\n");};};

/* Define B vector:
     For source being the ith node:
      B[e+i] = R = 1;
     For sinks being the jth nodes:
     B[e+j] = -R = -1;*/
double *B = (double *)calloc(t*(e+v_cnt), sizeof(double));

for(i=0;i<t;i++)
  {
    B[e*t+t*s+i] = R;
    B[e*t+t*sink1[i]+i] = -R;
  };
*Bp = B;
if (oDEBUG)
  {for(i=0;i<e*t+v_cnt*t;i++)
    printf("%f ", B[i]);
  printf("\n");
  };
for(i=0;i<e+(2*e*t);i++)
  {
    C[i] = 1;
  };

*v = v_cnt;
*count = counter
```

**APPENDIX C**

**The Main Algorithm**

This file takes the A_Z matrix as input arrays into the LP solver. The LP solver displayed the standard output base on the configurations at the command prompts

```
/*e: number of edge within the multicast group;
  v: number of nodes within the multicast group;
  t: number of sinks or receivers;
  d: packet delay;
  pl: packet loss;
  pr: packet reject;

   addr_b: first address of B vector
   addr_A_z
   addr_A_x */

int simp(int e, int t, int v, int d, int pl, int *C, double *B, int *A_z, int *A_x, double *A, int count)
{

  int ne = count;                    // ne == arrange the number of non-zero entries in the cost criterion
matrix

  int DEBUG =0;
  int oDEBUG =0;

  int i;                                  // i is a loop variable
  int num_rows, num_cols;            // num_rows = number of rows of the cost matix, num_cols =
number of   columns in the cost matrix
char var_name[(int) 2*(2*3)+3*4 +400*8];
  double Z,x2,x3,x4,x5,x6,x7,x8,x9,x10; // solution variables for the LP solver which are the number of
receivers;


  /* input arrays to the LP solver -- ia[], ja[], and ar[] */

  int *ia = (int *) calloc((ne+1), sizeof(int));
   glp_prob *lp;

  int *ja;
  double *ar;

  if ((ja = (int *) calloc((ne+1), sizeof(int))) == NULL){
    printf("Error");}

  if ((ar = (double *) calloc((ne+1), sizeof(double))) == NULL){
    printf("Error");}
  lp = glp_create_prob();

  glp_set_prob_name(lp, "improved network_coding");
  glp_set_obj_dir(lp, GLP_MIN);

num_rows = (int) t*(v+e)*8;

  glp_add_rows(lp, num_rows);

  for(i=1;i<=num_rows;i++)
    {
      sprintf(var_name, "B%i", i);
      glp_set_row_name(lp, i, var_name);
      glp_set_row_bnds(lp, i, GLP_FX, (double) *(B+i-1), (double) *(B+i-1));
    };
```

```
printf("\n");

num_cols = (int)2*(e*t)+e;
glp_add_cols(lp, num_cols);

for(i=1;i<=e;i++)
for(i=1; i<=4;i++)
  {
    sprintf(var_name, "x%i", i);
    glps_set_col_name(lp, i, var_name);
    glps_set_col_bnds(lp, i, GLP_DB, (double) 0.0, (double) *(C+i-1));
    glps_set_obj_coef(lp, i, (double) *(A+(i-1)));
    // printf("%s ", var_name);
  }
for(i=e+1;i<=e*t*8+e;400;i++)
  {
    sprintf(var_name, "x%i", i);
    glps_set_col_name(lp, i, var_name);
    glps_set_col_bnds(lp, i, GLP_LO, 0.0, 0.0);
    glps_set_obj_coef(lp, i, (double) *(A+(i-1)));
  }

for(i=e*t+e+1;i<=num_cols;i++)
  {
    sprintf(var_name, "x%i", i);
    glps_set_col_name(lp, i, var_name);
    glps_set_col_bnds(lp, i, GLP_DB, 0.0, (double)*(C+i-1));
    glps_set_obj_coef(lp, i, (double) *(A+(i-1)));
    //   printf("%s ", var_name);
  }


printf("\n");

//k =number of non-zero elements
//  e*2*t+ 2*e*t

int k=0, inx=1;

printf("e:%d t:%d, cols:%d, row:%d", e, t, num_rows, num_cols);


for(k=0;k<(ne-2*e*t)*3;k=k+3)
  {
    ia[inx] = *(A_z+k)+1;
    ja[inx] = *(A_z+k+1)+1;
    ar[inx] = *(A_z+k+2);
  //  if(DEBUG){printf("%d:%d %d %f\n", inx,*(A_z+k)+1, *(A_z+k+1)+1, (double) *(A_z+k+2));};
    inx++;
  };

                                    //printf("Yes\n");

                                   //   printf("\n DEBUG: ia = 2313, ja=6872ar[11432] = %f\n",  ar[11432]);
if(DEBUG)
  {for(i=1;i<=ne;i++)
    {printf("%d: %d %d %f\n", i, ia[i], ja[i], ar[i]);};};
else
  {for(j=1;i<=ne;j++)
    {printf("%d: %d %d %f\n", j, ja[j], ja[j], ar[j]);};};
```

```
  glp_load_matrix(lp, ne, ja, ia, ar);

  glp_simplex(lp, NULL);

  Z = glp_get_obj_val(lp);
  X2= glp_get_col_prim(lp, 2);
  X3= glp_get_col_prim(lp, 3);
  X4= glp_get_col_prim(lp, 4);
  X5= glp_get_col_prim(lp, 5);
  X6= glp_get_col_prim(lp, 6);
  X7= glp_get_col_prim(lp, 7);
  X8= glp_get_col_prim(lp, 8);
  X9= glp_get_col_prim(lp, 9);
  X10= glp_get_col_prim(lp, 10);


  printf("Z= %f X2 = %f X3 = %f X4 = %f X5=%f \n", (double) Z, (double)x2, (double)x3, (double)x4,
(double)x5);
    printf("X6 = %f X7 = %f X8 = %f X9 = %f X10= %f\n", (double)x6 ,(double)x7, (double)x8,
(double)x9, (double)x10);


    glp_delete_prob(lp); the LP solver delete the previous cost and keeps record of the present cost of
multicast for the receiver in question.

    return 0; // exit status == success;
}
```

This prints the cost based on the command prompts for each number of receivers depending on the number of randomly generated nodes considered.

**APPENDIX D**
**The Network Coding Algorithm**

This file contains the main algorithm which takes as argument the path of the input data file, and the size of the multicast group (the number of sinks) based on the randomly generated nodes

```
/*e: number of edge within the multicast group;
  v: number of nodes within the multicast group;
  t: number of sinks or receivers;
  d: packet delay (400ms);

* Define a source and sink nodes, chosen randomly through a random number generator*/

  int s, sink[3];   // 's' is the random number generated; sink[t] is an array of sink nodes;


  srand((unsigned)time(NULL));


  s = floor((((double) rand())/RAND_MAX) * (v_cnt));
  if (s >= v_cnt)
    printf("ERROR!!!: invalid random number generated");

 printf("s: %d\n", s);

else continue;
int main(int argc, char *argv[])
{
  //Customize: File Name and number of sinks 't'
  char filename[256];                                          // filename cannot be more than
256 characters long
  int t=1, t, t1;
  int wireless = 0;

  int j;                                          // loop variable

  for(j=0; j<argc; j++){
            if(strcmp(argv[j], "--file") == 0)
            {
             strcpy(filename, argv[++j]);
            }
            else if(strcmp(argv[j], "--sinks") == 0)
            {

             t = atoi(argv[++j]);
            }
            else if(strcmp(argv[j], "--wireless") == 0)
            {
            wireless = 1;
            }
            else continue;
}

  //Other Initialization
  int DEBUG=0;                                          // for debugging purposes; should be set to 0 for
normal operation


  /* variables for holding node names and edge costs, and the number of edges 'e' */
  double *data_ptr;
  int e;

  int v=-1;
```

```
 int i;                          // 'i' is a loop variable
 int count=0;                            // count keeps track of the total number of non-zero entries in the
cost matrix

 /* Call to filetest.c; reads the input file and stores its information in arrays. */
 filetest(filename, &data_ptr, &e);

 /* For debugging purposes; outputs information about the various variables if DEBUG = 1 */
 if(DEBUG)
  {
   for (i=0; i<e; i++)
        printf("%f %f %f \n", data_ptr[i*3], data_ptr[i*3+1], data_ptr[i*3+2]);

   printf("M=%f E=%d\n", data_ptr[0], e);};

printf("\nPerforming wireless optimization base on the problem formulated: \n");

 /* Allocation of memory for Link distances, packet delay and the B vector... */
 double * = (double *)calloc(e+2*e*t,sizeof(double));
 int *d = (int *) calloc((400)*e*t*3, value of packet delay (int));
 int * = (int *)calloc((d*e*t)*3, link distances (int));
 double *;
 int * = (int *) malloc((e+(2*e*t))*sizeof(int));


 /* call all the value of nodes distances, value of packet delay before simulation */
 formulate(data_ptr, e, t, d, &count, preforms wireless optimization using integer programming as
formulated);
/* If the inputs are incorrect, print usage information */

          {

           printf("\nFor wireless networks: \n");
           printf("      $ main --file [filename] --sinks [number of sinks (integer value)] --wireless\n");
           printf("              where $ is the command-line prompt.\n");

          exit(EXIT_FAILURE);
          }
          //printf("6.47584574548545\n");
 /* The following loop reads the command-line arguments passed into the main() method */


 /* display the result to standard output and delete previous record */
 printf("\nZ = %; X1 = %; X2 = %; X3 = %; X4 = %; X5 = %; X6 = %, X7 = %; X8 = %; .. Xn = %\n
 lpx_delete_prob(lp);


 return 0;                                  // exit status == success;
```

**APPENDIX E**
**Formation of A_Z Matrix (with three parameters)**

This file takes the network data in an array and the array contains all pertinent information about the nodes, their distances and the one additional constraints considered (packet delay, packet loss and packet rejection). This file extracts this information from the array and forms an A_Z matrix based on the wireless constraints formulated as mixed integer programming described in this research work.

/* the A_Z matric is formulated by the following method */

```
void formulate(double* data_ptr, int e, int t,int d, int  pl, int pr, double R, double* A, int* A_z, int*
A_x, double **Bp, int *C, int *v, int *t1, int *count, int wireless)
{

  /* Constraint Matrix:
     t: number of sinks,
     e: number of edges;
     v_cnt: number of nodes;
d: packet delay, (≤400)
     pl: packet loss; (≤4)
     pr: packet rejection, (≤8)

     |A_z| |X_z| = |b|
     |A_x| |X_x|          */

  /* Defining A_z Matrix (e*pr(t+1) by et)*pl
     This represents the additional variable constraint (packet loss and packet rejection) considered in
this research:
     z_e - x_e^t != 0*/


  //int *A_z = (int *) calloc((2e+1)*e*t*3*400, sizeof(int)); // e*(e+2)*t*4*8 elements; 3 entries for
each

  int inx = 0, cnt=0;

   (wireless)
  {
   printf("\nPerforming wireless optimization: \n");  the algorithm takes care of packet loss from zero
(0) to four (4), packet rejection of less than eight (8) and the value of packet delay of less than (400) for
maximum flow of packets during multicast.

  for(j=0;j<t*e;j=j+t)
   {
          Int_d [400];         // collect all indices with cost >= A[cnt]

          for(d=0; d<e; d++)
      {

           if(A[k] >= A[cnt]) {
              temp[d] = 400;
              temp[pr]=8;
              temp[pl]= 4;
        }
        else temp[d] = 0;
      }            // collected in temp


          for(i=0; i<t; i++)
    {
           for(d=0; d<e; d++)
           {
```

```
            if(temp[d] == 1 && list[2*d] == list[2*cnt])  // edge has cost >= and is emanating from the
same node
                {
            A_z[inx] = j+i;   // row j+i
                  A_z[inx+1] = d;   // column k, representing link z_(d+1)
            A_z[inx+2] = 1;   // set entry = 1
               //  printf("z(%d, %d) ", A_z[inx], A_z[inx+1]);

            A_z[inx+3] = j+i; // row j+i
                  A_z[inx+4] = e+e*t+d*t+((j+i)%t); // locate the appropriate x_d^t
            A_z[inx+5] = -1;  // set entry = -1
               //  printf("x(%d, %d) ", A_z[inx+3], A_z[inx+4]);

                        inx += 6;
              counter += 2;
                }
        }

              A_z[inx] = j+i;       // slack variable
              A_z[inx+1] = j+i+e;
              A_z[inx+2] = -1;
            //   printf("s(%d, %d)\n", A_z[inx], A_z[inx+1]);

            inx=inx+3;
       counter++;
     };             // for(i=0;...);
     cnt++;
   };               // for(j=0;...);
  }                 // (wireless)

 /* Defining A_x Matrix:
          v_cnt   : enter the number of nodes within the multicast group
               e    : enter the number of edges within the multicast group
               nodes[i]: list of nodes in an increasing order in the group
               list[i]: list of edges (nodeA->nodeB)
               size    : v_cnt by e

    Suppose *NodeA->NodeB edge is ith edge in the slist,
          *Node A is pth in the nodes of the multicast group,
          *Node B is qth in the nodes. Then,

    pth row ith col = 1 (outgoing)
    qth row ith col = -1 (incoming)
    This matrix efficiently represents the flows of packets (messages) within the multicast group:
 */

 //int *A_x = (int *)calloc((2*e*t)*3, sizeof(int));

 inx = 0;

 for(i=0; i<e*2; i=i+2)
   {
     for(j=0;j<v_cnt;j++)
           {
             if(list[i] == nodes[j]) {p = j;}
             if(list[i+1] == nodes[j]){q = j;}
           }

     for(k=0;k<t;k++)
           {
```

```
            A_x[inx] = (e*t)+t*p+k;
            A_x[inx+1] = (e*(t+1))+ t*(i/2)+k;
            A_x[inx+2] = 1;

            A_x[inx+3] = (e*t)+t*q+k;
            A_x[inx+4] = (e*(t+1))+t*(i/2)+k;
            A_x[inx+5] = -1;

            inx= inx+6;
        counter += 2;
           };
  };

  for(i=0; i<2*e; i=i+2)
  {
   printf("%d %d \n", list[i], list[i+1]);
  };
//Testing
if (oDEBUG)
  {
    printf("%s\n", "List");
    for(i=0;i<2*e;i=i+2)
          {printf("%d %d \n", slist[i], slist[i+1]);};};

if (oDEBUG)
  {
    printf("%s\n", "Node List:");
    for (i=0;i<v_cnt;i++)
          {printf("%d: %d\n", i, nodes[i]);};
    printf("\n");
  };

if(oDEBUG)
  {for(i=0;i<2*e*t*3/6;i++)
    {for(j=0;j<6;j++)
          {printf("%d ", A_x[i*6+j]);}
    printf("\n");};};

/* Define B vector:
     For source being the ith node:
      B[e+i] = R = 1;
     For sinks being the jth nodes:
     B[e+j] = -R = -1;*/
double *B = (double *)calloc(t*(e+v_cnt), sizeof(double));

for(i=0;i<t;i++)
  {
    B[e*t+t*s+i] = R;
    B[e*t+t*sink1[i]+i] = -R;
  };
*Bp = B;
if (oDEBUG)
  {for(i=0;i<e*t+v_cnt*t;i++)
    printf("%f ", B[i]);
  printf("\n");
  };
for(i=0;i<e+(2*e*t);i++)
  {
    C[i] = 1;
  };
```

```
*v = v_cnt;
*count = counter;
```

**APPENDIX F**

**Test file of the Randomly Generated Nodes by the MIPA**

Node 0 Node 1000 0.34053156537931056

Node 0 Node 3000 3.5038322244172155

Node 0 Node 12000 6.79400957192307

Node 0 Node 14000 7.028767318599552

Node 0 Node 15000 1.3805867394238394

Node 1000 Node 0 0.34053156537931056

Node 1000 Node 3000 1.7795582618848038

Node 1000 Node 12000 4.8284816380010955

Node 1000 Node 15000 0.7214826715926195

Node 2000 Node 4000 3.2135627998481904

Node 2000 Node 5000 1.2739861707654374

Node 2000 Node 9000 1.223419886286729

Node 2000 Node 10000 7.166681332413034

Node 2000 Node 14000 6.823701106122628

Node 2000 Node 19000 4.341390777549075

Node 3000 Node 0 3.5038322244172155

Node 3000 Node 1000 1.7795582618848038

Node 3000 Node 6000 5.913054798710136

Node 3000 Node 12000 5.38532033446869

Node 3000 Node 15000 2.8374027267657587

Node 3000 Node 17000 7.537138422044333

Node 4000 Node 2000 3.2135627998481904

Node 4000 Node 5000 5.466733513681688

Node 4000 Node 8000 5.6532025181074985

Node 4000 Node 9000 1.6392492136318422

Node 4000 Node 10000 2.3819966833917623

Node 4000 Node 13000 2.9503442240967264

Node 4000 Node 14000 3.5124320036792227

Node 4000 Node 16000 8.797384414894498

Node 5000 Node 2000 1.2739861707654374

Node 5000 Node 4000 5.466733513681688

Node 5000 Node 9000 4.640642308893887

Node 5000 Node 14000 4.987171972980809

Node 5000 Node 15000 8.636064615661134

Node 5000 Node 19000 5.329569599851498

Node 6000 Node 3000 5.913054798710136

Node 6000 Node 12000 8.231873148278444

Node 7000 Node 11000 5.521089329570453

Node 8000 Node 4000 5.6532025181074985

Node 8000 Node 13000 5.562281260983681

Node 8000 Node 14000 1.5228700267923618

Node 8000 Node 16000 0.3674009325651853

Node 8000 Node 18000 8.699501641480762

Node 9000 Node 2000 1.223419886286729

Node 9000 Node 4000 1.6392492136318422

Node 9000 Node 5000 4.640642308893887

Node 9000 Node 10000 2.5883571367212577

Node 9000 Node 13000 7.550786096539746

Node 9000 Node 14000 8.144817651828369

Node 9000 Node 19000 8.165618326314661

Node 10000 Node 2000 7.166681332413034

Node 10000 Node 4000 2.3819966833917623

Node 10000 Node 9000 2.5883571367212577

Node 10000 Node 13000 2.972958006751024

Node 11000 Node 7000 5.521089329570453

Node 11000 Node 12000 2.746642178813417

Node 11000 Node 15000 8.893268264265258

Node 12000 Node 0 6.79400957192307

Node 12000 Node 1000 4.8284816380010955

Node 12000 Node 3000 5.38532033446869

Node 12000 Node 6000 8.231873148278444

Node 12000 Node 11000 2.746642178813417

Node 12000 Node 15000 2.0551317847970085

Node 13000 Node 4000 2.9503442240967264

Node 13000 Node 8000 5.562281260983681

Node 13000 Node 9000 7.550786096539746

Node 13000 Node 10000 2.972958006751024

Node 13000 Node 14000 7.480509203547326

Node 13000 Node 16000 8.47779211892295

Node 14000 Node 0 7.028767318599552

Node 14000 Node 2000 6.823701106122628

Node 14000 Node 4000 3.5124320036792227

Node 14000 Node 5000 4.987171972980809

Node 14000 Node 8000 1.5228700267923618

Node 14000 Node 9000 8.144817651828369

Node 14000 Node 13000 7.480509203547326

Node 14000 Node 16000 2.4766488584535638

Node 15000 Node 0 1.3805867394238394

Node 15000 Node 1000 0.7214826715926195

Node 15000 Node 3000 2.8374027267657587

Node 15000 Node 5000 8.636064615661134

Node 15000 Node 11000 8.893268264265258

Node 15000 Node 12000 2.0551317847970085

Node 16000 Node 4000 8.797384414894498

Node 16000 Node 8000 0.3674009325651853

Node 16000 Node 13000 8.47779211892295

Node 16000 Node 14000 2.4766488584535638

Node 16000 Node 18000 5.824909225643624

Node 17000 Node 3000 7.537138422044333

Node 18000 Node 8000 8.699501641480762

Node 18000 Node 16000 5.824909225643624