

**NETWORK CODING ASSISTED MULTICASTING IN MULTI-HOP  
WIRELESS NETWORKS**



*Prateek Rathore*



# **NETWORK CODING ASSISTED MULTICASTING IN MULTI-HOP WIRELESS NETWORKS**

A

*Thesis Submitted*

*in Partial Fulfillment of the Requirements*

*for the Degree of*

**DOCTOR OF PHILOSOPHY**

By

**PRATEEK RATHORE**



DEPARTMENT OF ELECTRONICS AND ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

GUWAHATI – 781039, ASSAM, INDIA

August, 2020



## Certificate

This is to certify that the thesis entitled “**NETWORK CODING ASSISTED MULTICASTING IN MULTI-HOP WIRELESS NETWORKS**”, submitted by **Prateek Rathore** (136102015), a research scholar in the *Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by him under our supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated:  
Guwahati.

Dr. Kalpana Dhaka  
Assistant Professor  
Dept. of Electronics and Electrical Engg.  
Indian Institute of Technology Guwahati  
Guwahati - 781 039, Assam, India.

Dated:  
Guwahati.

Prof. Sanjay K. Bose  
Professor  
Dept. of Electronics and Electrical Engg.  
Indian Institute of Technology Guwahati  
Guwahati - 781 039, Assam, India.



*Dedicated To My Beloved Family*

*And*

*The Almighty God*



## **Acknowledgements**

First and foremost, I take this opportunity to thank God for blessing me to learn and pursue my passion. I would have never achieved this accomplishment without the faith I have in the Almighty God. I feel immense pleasure in expressing my sincere gratitude to my thesis supervisors Dr. Kalpana Dhaka and Prof. Sanjay K. Bose. I would always be indebted to them to nurture my scientific skills and carry out a thorough review of my so many drafts for correcting my ideas and language with great patience. I have learned a lot from their dedication to the work and consistency of excellence in research methods, which I will cherish throughout my life. I would sincerely thank my committee members Prof. Ratnajit Bhattacharjee, Dr. A. Rajesh, and Dr. Ramesh K. Sonkar, for their thoughtful discussions and constructive comments, which have immensely influenced my research work.

I am grateful to MHRD, Government of India, and IIT Guwahati to provide me with research fellowship and travel support for attending the conferences during my research work. I am thankful to Arijit, Rahul, Ramanand, Vijith, Mohit, Pawan, Anoop, Sonali, Mog, Kaushik, and Shalini for discussions and keeping the lab friendly. I want to say my special thanks to my IITG friends, Birjit, Achyut, Aditya, Nishant, Ashish, Vishal, Sunil, Tushar, Dipanker, Subhashish, Niladri, and other M.Tech., and Ph.D. students, from whom I, directly and indirectly, benefited over the past few years. I also would like to thank my dearest friends Mahendra K. Shukla and Om Jee Pandey, for their support, inspiration, and friendship outside the academic world.

Last but not least, I would like to express my deepest gratitude, love, and affection to my parents and a younger brother who supported me throughout the long path of my academic endeavor and endured countless sacrifices. To them, I owe all I have ever achieved.

***Prateek Rathore***



## **Abstract**

Multicasting is the process of disseminating the same information to multiple destinations as in file distribution, distance learning, and video-on-demand services. Multicasting applications are broadly classified based on whether they require reliable or unreliable transmissions. In unreliable applications like on-demand video transmissions or live video streaming, high video quality within a predefined threshold time is required. The hard deadline for packet reception is achieved for all the clients with some packet loss. In reliable multicasting applications like a software update or downloading a medical image, strict packet reliability is needed at the receivers. The primary objective is to communicate all the packets to the destination nodes with efficient network resource utilization. This thesis focuses explicitly on reliable multicasting where the source and destination nodes are deployed in a multi-hop wireless network scenario.

In the existing multicasting schemes, the wireless links are independent of each other, and the proposed protocols aim specifically on reducing delays, bandwidth utilization, energy consumption, and expected transmission count. Multicasting in wireless networks involves many receiving nodes, and their links may not always be completely independent of each other. Link correlation may then have an impact on multicasting performance. This thesis considers the correlation between the links while designing the multicasting protocols. The first contribution is to determine the multicasting cost and the associated forwarder set while taking links correlation into account.

Network Coding is used in multicasting to combine multiple lost packets and send them in a single transmission. The second contribution demonstrates the application of network coding in multicasting over a multi-hop wireless network. The packets are efficiently transmitted from the source node to the corresponding destination set using the selected

forwarder nodes to reduce the ETXs for efficient network capacity usage. Network coded packets are transmitted by allocating traffic efficiently among the forwarding nodes while ensuring that each destination receives all packets. Extensive simulation results are presented for both cases when the packets receptions over the different links are independent or correlated. The proposed protocol significantly reduces the expected number of overall transmissions in the network, for both independent links and correlated links, as compared to the existing multicasting schemes.

In a random wireless network, the wireless broadcasting feature can combine packets of different sessions, which are simultaneously going on in the network. Due to the broad applications of multicasting, there are higher chances of having multiple-multicasting sessions in the network. The third contribution of this thesis introduces a network coding assisted reliable multi-source multicasting technique for multi-hop wireless networks. The selected forwarding nodes are used to efficiently transmit the packets from multiple source nodes to their corresponding destination sets. A CW is defined at each intermediate node, and the packets present in it are judiciously combined. These packets are efficiently distributed among the neighboring nodes so that the destinations connected to each source receive the desired packets. The primary objective is to reduce the multicasting cost expressed in terms of the average number of packet transmissions over the network. This in turn reduces resource expenditure and tackles the traffic congestion problem of the network. A significant reduction is found in the overall transmission cost of the proposed scheme for both correlated and independent link networks.





# Contents

<b>List of Figures</b>	<b>xix</b>
<b>List of Tables</b>	<b>xxi</b>
<b>List of Symbols</b>	<b>xxiii</b>
<b>List of Abbreviations</b>	<b>xxvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Wireless Correlated Links	2
1.2 Network Coding	3
1.3 Reliable and Unreliable Multicast	4
1.4 Motivation and Research Objectives	5
1.5 Thesis Outline and Contributions	8
<b>2 Literature Survey: Review of Related Work</b>	<b>11</b>
2.1 Wireless Networks with Correlated Links	12
2.2 Reliable Multicast in Wireless Networks	15
2.3 Unreliable Multicast in Wireless Networks	21
2.4 Multi-Source Multicast in Wireless Networks	25
2.5 Summary	28
<b>3 Multicasting in Wireless Networks with Correlated Links</b>	<b>31</b>
3.1 Introduction	31
3.2 Multicasting over a Network with Correlated Links	32
3.3 Multicasting Route Cost Calculation with Correlated Links	35
3.4 Packet Transmission Strategy	42
3.5 Multicasting with Correlated Links: Example	42
3.6 Summary	44

<b>4 Network Coding Assisted Reliable Multicasting in Multi-Hop Wireless Networks</b>	<b>47</b>
4.1 Introduction	47
4.2 Implementation Details: NC-MLCAR	48
4.3 Proposed Algorithm: NC-MLCAR	51
4.4 Simulation Results and Complexity Analysis	61
4.4.1 Simulation Results	61
4.4.2 Complexity Analysis	63
4.5 Summary	65
<b>5 Network Coding Assisted Reliable Multi-Source Multicasting in Multi-Hop Wireless Networks</b>	<b>67</b>
5.1 Introduction	67
5.2 Implementation Details: NCRMM	68
5.3 Proposed Algorithm: NCRMM	72
5.4 Simulation Results and Complexity Analysis	83
5.4.1 Simulation Results of TSMNC	84
5.4.2 Simulation Results of NCRMM	85
5.4.3 Complexity Analysis	89
5.5 Summary	91
<b>6 Conclusions and Future Scopes</b>	<b>95</b>
6.1 Conclusions	95
6.2 Future Scopes	97
<b>References</b>	<b>101</b>
<b>List of Publications</b>	<b>111</b>





# List of Figures

1.1	Example network: Network coding	3
3.1	Example network 1 with correlation between the links	34
3.2	Example network 2: Multicasting	37
3.3	Venn diagram showing correlation between the links of the example network 2	39
3.4	Example network 3: Multicasting	43
4.1	Flowchart of NC-MLCAR	50
4.2	Example network 1: Multicasting	52
4.3	Example network 2: Multicasting	60
4.4	Average multicasting cost of different schemes for independent and correlated links	62
5.1	Flowchart of NCRMM	73
5.2	Example network 1: Multi-source multicasting	74
5.3	Example network 2: Multi-source multicasting	81
5.4	Multicasting cost of 10 random networks along with their average cost	85
5.5	Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in independent link scenario	86
5.6	Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in low link correlation scenario	87
5.7	Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in high link correlation scenario	88
5.8	Average multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes	89



## List of Tables

3.1	Comparison of multicast route cost and selected forwarder set for an example network 3 having independent links and correlated links	43
4.1	Comparison of complexity analysis of MLCAR and NC-MLCAR	64
5.1	Comparison of complexity analysis of MLCAR, NC-MLCAR, and NCRMM	91





# List of Symbols

$NN$	set of nodes in the network
$n$	a network node
$m_s$	number of multicasting sessions in the network
$q$	a multicasting session, $q \in \{1, \dots, m_s\}$
$n_m$	set of multicasting sessions in the network, $n_m = (1, \dots, m_s)$
$ \cdot $	denotes cardinality of the set
$rand$	uniformly distributed random number $\in (0, 1)$
$S^{(q)}$	source node of the $q^{\text{th}}$ session
$DS^{(q)}$	destination set of the $q^{\text{th}}$ session
$FS^{(q)}$	forwarder set of the $q^{\text{th}}$ session
$NS^{(q)}$	neighboring node of $S^{(q)}$
$DS_n^{(q)}$	destination set reached by node $n$
$FS_n^{(q)}$	set of potential forwarder nodes of node $n$ reaching $DS_n^{(q)}$
$NS_n$	set of neighboring nodes of node $n$ in which each node covers at least one node from $DS_n^{(q)}$
$f, f_u, f_v, f_p$	a node of $FS_n^{(q)}$
$w$	$w \in \{i, c\}$ , where $i$ and $c$ stands for independent and correlated link networks, respectively
$d$	a node of $DS_n^{(q)}$
$f_n^{(q),c}$	subset of $FS_n^{(q)}$ which reliably communicates a packet from node $n$ to $DS_n^{(q)}$
$DS_{f_n^{(q),c}}^{(q)}$	subset of $DS_n^{(q)}$ reached by $f_n^{(q),c}$
$f_b$	subset of $FS_n^{(q)}$

$D_{f_u}$	subset of $DS_n^{(q)}$ reached by $f_u$
$f_x^{(q),w}$	subset of neighboring nodes of $S^{(q)}$ reaching $DS^{(q)}$
$f_a^{(q),w}$	subset of $f_x^{(q),w}$ which gets the packet
$f_y^{(q),w}$	subset of $f_x^{(q),w}$ which has not received the packet
$f_{aa}^{(q),c}$	actual forwarders which are subset of $f_a^{(q),c}$
$DS_{f_a^{(q),w}}^{(q)}$	subset of $DS^{(q)}$ reached by $f_a^{(q),w}$
$DS_{f_y^{(q),w}}^{(q)}$	subset of $DS^{(q)}$ reached by $f_y^{(q),w}$
$\beta_{f_a^{(q),w}}^{(q),w}$	probability of packet reception at each node in $f_a^{(q),w}$ and none of the nodes in $f_y^{(q),w}$
$P_F$	probability that packet does not reach at any node in $f_x^{(q),i}$
$C_{S^{(q)}, DS^{(q)}}^{(q),w}$	minimum multicasting cost of transmitting a packet from $S^{(q)}$ to all the nodes in $DS^{(q)}$
$C_{S^{(q)}, DS_{f_y^{(q),w}}^{(q)}}^{(q),w}$	minimum multicasting cost of transmitting a packet from $S^{(q)}$ to all the nodes in $DS_{f_y^{(q),w}}^{(q)}$
$C_{f_a^{(q),w}, DS_{f_a^{(q),w}}^{(q)}}^{(q),w}$	minimum multicasting cost of transmitting a packet from $f_a^{(q),w}$ to all the nodes in $DS_{f_a^{(q),w}}^{(q)}$
$C_{x,y}^{(q),w}$	minimum multicasting cost to transmit a packet from node $x$ to destination subset $y$
$C_{opt}^{(q),c}$	optimal multicasting cost from $f_a^{(q),c}$ to $DS_{f_a^{(q),c}}^{(q)}$
$C_{n, DS_n^{(q)}, DS_{f_a^{(q),c}}^{(q),copy}}$	variables
$DS_{us}$	
$F_n^{(q)}$	set formed by subsets in $FS_n^{(q)}$
$D_n^{(q)}$	set formed by subsets in $DS_n^{(q)}$
$IP_n^{(q)}$	innovative packets allocated to the elements of $F_n^{(q)}$

$DE_{f_u}^{(q)}$	set of nodes of $D_n^{(q)}$
$PE_{f_u}^{(q)}$	set of innovative packets of $IP_n^{(q)}$
$N^{(q)}$	number of innovative packets of the $q^{\text{th}}$ session
$N_n^{(q)}$	number of innovative packets of the $q^{\text{th}}$ session present at node $n$
$B^{(q)}$	batch of innovative packets of the $q^{\text{th}}$ session in $S^{(q)}$
$B_n^{(q)}$	batch of innovative packets of the $q^{\text{th}}$ session in network node $n$
$M^{(q)}$	size of $DS^{(q)}$
$M_n^{(q)}$	size of $DS_n^{(q)}$
$V_n^{(q)}$	size of $FS_n^{(q)}$
$FST_d^{(q)}$	subset of $FS_n^{(q)}$ reaching $d, d \in DS_n^{(q)}$
$AFST^{(q)}$	set of $FST_d^{(q)}$ 's arranged in ascending order of size
$ENE^{(q)}$	set of subsets of $FS_n^{(q)}$ which must receive all innovative packets of the $q^{\text{th}}$ session that are transmitted from node $n$ to $DS_n^{(q)}$
$K^{(q)}$	size of $ENE^{(q)}$
$EFSN$	set of $ENE^{(q)}$ 's
$V_E$	size of $EFSN$
$N_{cw}$	number of innovative packets present in CW of a node
$IPCW_n$	innovative packets present in CW of node $n$
$NE_n$	set of nodes in $EFSN$ which are neighbors of node $n$
$FSU^{(q)}$	set of subsets of $FS_n^{(q)}$ which reaches $DS_n^{(q)}$
$DSU_{f_p}^{(q)}$	subset of $DS_n^{(q)}$ reached by node $f_p$



# List of Abbreviations

ACK	Positive Acknowledgment
BL	Base Layer
BS	Base Station
CF	Collective Flooding
CH	Cluster Head
CTS	Clear To Send
CW	Coding Window
D2D	Device to Device
DCN	Data Center Network
EL	Enhancement Layer
EMTX	Expected Multicast Transmission Count
ETX	Expected Number of Transmission
IoT	Internet of Things
LCAR	Least Cost Anypath Routing
MAC	Medium Access Control
MANET	Mobile Ad-hoc Network
MLCAR	Multicast Least Cost Anypath Routing
MORE	MAC-independent Opportunistic Routing Protocol
NACK	Negative Acknowledgement
NC	Network Coding
NCMLCAR	NC assisted Multicast Least Cost Anypath Routing
NCRMM	NC assisted Reliable Multi-source Multicasting
OR	Opportunistic Routing

QoS	Quality of Service
RLNC	Random Linear NC
RTS	Ready To Send
SDN	Software Defined Networking
SINR	Signal to Interference plus Noise Ratio
SLNC	Secure Linear NC
SVC	Scalable Video Coding
SVT	Scalable Video Transmission
TSMNC	Two-Source Multicasting using NC
TT	Transmission Topology





# Chapter 1

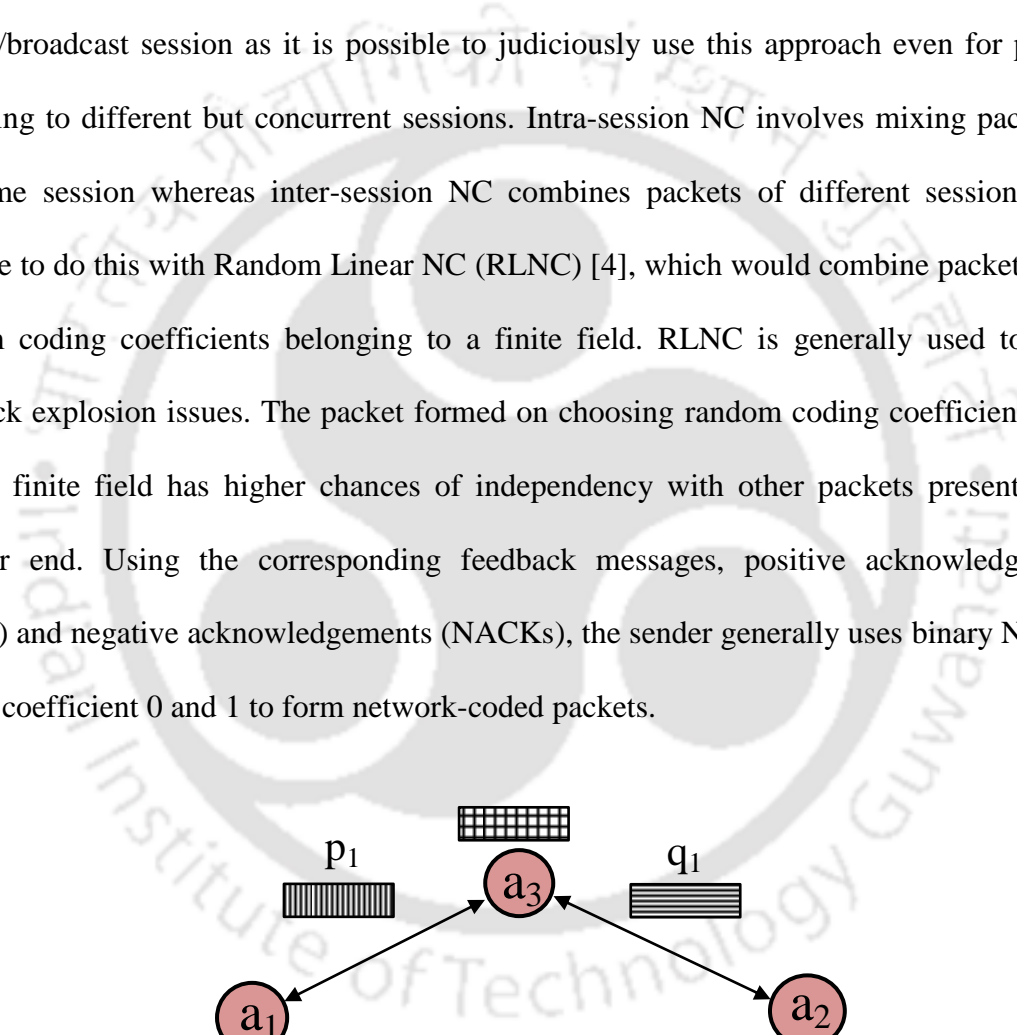
## Introduction

Over the past few years, a major portion of internet traffic has moved on to the wireless medium. Wireless communication is widely employed in applications such as online TV, health care services, home automation and video conferencing. Traditionally, the primary mode of communication has been *unicasting* with one-to-one transmission as, for example, in a client-server exchange. *Broadcasting* is a one-to-all transmission as in radio and television communication systems where a source node transmits to all receivers within range. In this thesis, the focus is on *multicasting* which is a one-to-many (but may not be to all) transmission mode combining attributes of both unicasting and broadcasting for one-to-many communications. Therefore, multicasting is the process of distributing the same information to multiple receivers present in a wireless network. Many subscription services like music club subscriptions or newspaper subscriptions would require the same information to be sent to all the requesting clients. Distributing latest version of the software as demanded by a number of users would require efficient multicast transmission over the network. Similarly, many sales representatives may use multicasting to advertise their products simultaneously to several customers present in different parts of the world. Other applications which would need multicasting support are distance learning, file distribution and video-on-demand services. Looking at the broad demand for these services, it is not surprising that multicasting becoming increasingly popular to support this type of group communication using the network resources as efficiently as possible.

## 1.1 Wireless Correlated Links

Due to the broadcast nature of wireless transmissions, any node within the coverage range of the sender can receive the packet and transmit it further [1], [2]. Packet dissemination over the wireless medium is error-prone and therefore, it is desirable to exploit the inherently broadcast nature of wireless links as much as possible while designing multicasting protocols which use network resources efficiently. The design of multicasting protocols depends on the wireless link cost and the correlations between the neighboring links. The Expected Number of Transmissions (*ETX*) required to reliably communicate a packet over the link is a key metric used to determine wireless link cost. The neighboring links in a wireless network are considered to be independent of each other, if the packet reception at the associated receive nodes is uncorrelated. This type of independence has typically been what has been assumed for the links in wireless networks. However, this assumption does not hold in many practical scenarios where the neighboring links do have some finite correlation between themselves even when the *ETX* cost of individual links remain the same [3]. The links of a network are considered to be highly correlated if the receiving nodes have higher chances of successfully getting the packet in the same transmissions. When links have a low correlation, the probability of the packet traversing the links correctly in the same transmissions would be less. The advantage offered in wireless broadcasting would therefore depend on the extent of correlation between the neighboring links. This would not only affect the multicasting cost but also the choice of the forwarding nodes used for packet transmissions. Therefore, taking link correlation into account may be expected to lead to more efficient multicast routing protocols.

## 1.2 Network Coding

The inherent advantage of a wireless broadcast network is that it provides excellent opportunities to combine packets of different receivers and send it forward in a single transmission. Network Coding (NC) can then use this to improve the overall throughput and delay performance of the network. The combined packets do not have to be from the same unicast/broadcast session as it is possible to judiciously use this approach even for packets belonging to different but concurrent sessions. Intra-session NC involves mixing packets of the same session whereas inter-session NC combines packets of different sessions. It is possible to do this with Random Linear NC (RLNC) [4], which would combine packets using random coding coefficients belonging to a finite field. RLNC is generally used to avoid feedback explosion issues. The packet formed on choosing random coding coefficients from a large finite field has higher chances of independency with other packets present at the receiver end. Using the corresponding feedback messages, positive acknowledgements (ACKs) and negative acknowledgements (NACKs), the sender generally uses binary NC with coding coefficient 0 and 1 to form network-coded packets. 

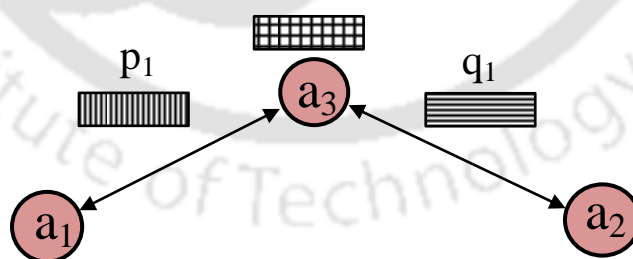


Figure 1.1: Example network: Network coding

As a simple example of NC, consider the network of Figure 1.1, where the wireless links are assumed to be perfect. Node  $a_1$  sends its packet  $p_1$  to the destination node  $a_2$  and node  $a_2$  sends its packet  $q_1$  to the destination node  $a_1$ , node  $a_3$  acts as a relay node for the two cases.

The two source nodes require four timeslots to transmit packets to the desired destinations. In case of NC,  $a_1$  first sends  $(p_1)$  to  $a_3$ ,  $a_2$  sends  $(q_1)$  to  $a_3$ , node  $a_3$  performs simple XOR operation of the two packets,  $(p_1 \oplus q_1)$  and broadcasts it to both  $a_1$  and  $a_2$ . Node  $a_1$  can extract  $(q_1)$  using  $(p_1)$  and  $(p_1 \oplus q_1)$  and node  $a_2$  can extract  $(p_1)$  using  $(q_1)$  and  $(p_1 \oplus q_1)$ . Thus, a total of three timeslots are required, which saves one timeslot compared to the case when NC is not employed. NC increases the transmission efficiency by combining missing packets into one coded packet where intra-session NC is used to improve the throughput of a single session while inter-session NC operates over multiple sessions and increases the overall throughput of the network.

### **1.3 Reliable and Unreliable Multicast**

Multicasting applications are broadly classified based on their requirement for reliable or unreliable delivery to their intended destinations. In reliable multicasting for applications like software updates and downloading a new security patch, strict packet reliability is needed at all the intended receivers. The feedback messages, i.e. ACKs and NACKs, are used to ensure reliable delivery of all the packets to each member of the destination set. The primary objective of reliable multicasting is to send all the intended packets to the destined destination nodes with as efficient network resource utilization [5] as possible. Unreliable multicasting may be sufficient for applications like on-demand video transmissions and live video streaming which requires packet delivery within a predefined threshold time; packets which are not delivered within this time are typically discarded and are not used at the destination. In typical network, destination users and intermediate forwarders would be randomly located, and the aim would be determining the minimum cost multicasting path [6] which delivers the data within a specific upper-bounded end-to-end delay. The specific details of the

multicasting approach used would greatly depend on the nature of the applications that are required to be supported.

#### **1.4 Motivation and Research Objectives**

This thesis focuses on reliable multicasting over multi-hop wireless networks. Reliable multicasting addresses emerging applications where strict packet reliability is maintained as in file distribution, popular file downloads, billing records, electronic newspaper, and medical images. A random wireless network may have multicasting clients present anywhere in the network. Therefore, the critical challenge is to send all the packets to the desired destinations by utilizing network resources efficiently in any random topology. Most of the existing multicasting schemes assume that the wireless links are independent of each other, and the proposed protocols aim specifically on reducing delays, bandwidth utilization, energy consumption, and expected transmission count. However, multicasting in wireless networks involves many receiving nodes, and the links used to reach them may not be completely independent of each other. The correlation between the wireless links has an impact on the performance metrics and therefore, should be taken into consideration while designing the multicasting protocols.

NC can be used in multicasting to combine multiple lost packets into one coded packet and communicate it to the intended receivers. A coded packet is considered innovative if it is linearly independent of the packets already present at the receiving node. Various authors have earlier considered NC-assisted multicasting protocols but these have usually been specific to particular types of networks and it would be difficult to generalize them to any random multi-hop wireless network. Other multicasting protocols proposed for multi-hop wireless networks using RLNC, like Medium Access Control (MAC)-independent

Opportunistic Routing Protocol (MORE) [4] and Pacifier [7], randomly mix the packets for forwarding to the neighboring nodes but have fairly complicated encoding and decoding structures. Using RLNC in a multi-hop wireless network may cause unnecessary packet transmissions and lead to traffic congestions and inefficient bandwidth utilization [4]. This can be overcome with proper use of ACKs and NACKs to implement binary NC, which has simpler coding and decoding structures rather than RLNC schemes.

In a random wireless network, the wireless broadcasting feature has the advantage of also being able to combine packets of different sessions, which are going on in the network simultaneously. Due to the broad applications of multicasting, there are higher chances of having multiple-multicasting sessions in the network. A straightforward way of communicating packets from each source to its respective destinations is to apply the protocols defined for single-source multicasting scenarios directly. However, this scheme may result in high overhead and inefficient data transmission.

The specific objectives of the thesis are

- Formulate a generalized expression to calculate multicasting cost so as to determine the associated forwarder set having least cost in a correlated link environment.
- Propose a novel NC assisted multicasting protocol in multi-hop wireless network applicable for both independent and correlated links.
- Show the impact of link correlation on NC where the packets are judiciously combined such that the network-coded packet formed is innovative for the maximum number of receiving nodes.

- Propose a new NC assisted multi-source multicasting protocol in multi-hop wireless network such that the overall throughput is improved for both independent and correlated link networks.

The thesis presents a comprehensive study of reliable multicasting in the multi-hop wireless network. The impact of correlation between the wireless links on the multicasting cost is studied and the corresponding forwarder set associated with it is determined. A generalized expression is proposed to determine the least-cost multicasting path that can be used to transmit packets efficiently from the source node to the designated destination set. Incorporation of NC further improves the multicasting efficiency by reducing the expected number of overall transmissions in the network. A NC assisted multicasting protocol is proposed in which packet reception is tracked using ACKs and NACKs. The scheme ensures that all the innovative packets are efficiently forwarded to the successive nodes so as to ultimately reach the destination set, improving the overall network performance. Since NC can also take into account the information content of several sessions which may be simultaneously present in the network, it can be used to further improve the overall throughput of the network. NC scheme is proposed for the multi-source multicasting scenarios where intermediate nodes store packets of different sessions and can therefore perform both intra-session and inter-session NC. The packet reception at the neighboring nodes are communicated using ACKs and NACKs which ensure that the innovative packets of all the multicasting sessions are transmitted to forwarding nodes which can then send them further towards their associated destination sets. With the help of NC, the expected number of overall transmissions in the network can then be reduced. This would, in turn, ensure that resources are used efficiently and would better tackle delay and congestion problems in the network.

## 1.5 Thesis Outline and Contributions

The proposed multicasting protocols consider a multi-hop wireless network with independent and correlated links. The impact of link correlation in multicasting is shown which is applicable in a real-life scenario. NC assisted transmission protocols are presented which improve the network performance by reducing the multicasting cost. For the proposed protocol, a Coding Window (CW) is introduced at the intermediate nodes where packets of different multicasting sessions are stored and combined judiciously before communicating them to their neighboring nodes. The contributions of this thesis are listed below.

- In **Chapter 2**, a thorough discussion on the literature work related to reliable, unreliable, and multi-source multicasting is presented. The role of NC in improving the network throughput in different scenarios is also discussed in the chapter.
- An analytical expression for cost calculation in case of multicasting over correlated wireless links is proposed in **Chapter 3**. The number of transmissions needed to successfully communicate a packet over a network with correlated links is different from that of a network where links are assumed independent, considering the link costs to be the same in both the cases. The cost metric of correlated links is computed using the cost for direct transmission and correlation between the neighboring links. The Multicast Least Cost Anypath Routing (MLCAR) scheme [5] is used to identify the least cost route for networks with known link correlations and for a network with no information on link correlations and the results are compared. The simulation results show that the multicasting cost and the corresponding routes differ significantly between the two.

- **Chapter 4** demonstrates the application of NC in multicasting over a multi-hop wireless network. The forwarder set is selected using the MLCAR algorithm [5]. The packets are efficiently transmitted from the source node to the corresponding destination set using a forwarder set selected to reduce the ETXs for efficient network capacity usage. Network-coded packets are transmitted by allocating traffic efficiently between the nodes of the forwarder set while ensuring that each destination receives all the packets. Extensive simulation presented for the cases when the packets receptions over different links are considered to be both independent and correlated. The proposed protocol significantly reduces the expected number of overall transmissions in the network, for both independent links and correlated links, as compared to the existing multicasting schemes.
- In **Chapter 5**, a NC assisted reliable multi-source multicasting (NCRMM) technique is presented for multi-hop wireless networks. The MLCAR algorithm is used to select the neighboring nodes participating in packet transmission from multiple source nodes to the corresponding destination sets. A CW is defined at each intermediate node and the packets present in it are judiciously combined. These packets are efficiently distributed among the neighboring nodes so that the destinations for each source can ultimately receive the desired packets. Our primary objective is to reduce the overall multicasting cost of the network expressed in terms of the average number of packet transmissions from multiple sources to their respective destination sets. The simulation results are presented for both independent and correlated link networks.
- **Chapter 6** concludes this thesis with suggestions on future directions which can be investigated.



# Chapter 2

## Literature Survey: Review of Related Work

In this chapter, the major research works, which are publicly available, dealing with multicasting in wireless networks, is briefly described. Due to the inherent broadcast capability (full or limited) of wireless networks, a packet communicated from the source may be received simultaneously by multiple receivers in its coverage area. This broadcast advantage may reduce overall energy consumption and the use of network resources and would also help to lower traffic congestion in the network [1] - [5]. However, correlations between the neighboring links significantly impact the broadcast advantage and performance of the network. In Section 2.1, research work considering the correlation between the links is discussed. The research work related to single-source, reliable multicasting using wireless broadcast are discussed in Section 2.2. Similarly, Section 2.3 presents various schemes which provide unreliable multicasting from a single source to the corresponding destinations. The throughput of these and other similar schemes may be improved by using *Network Coding* (NC) where multiple packets, either from the same session or from multiple sessions/sources, are combined and transmitted. *Intra-session* NC refers to the case where packets of the same session are combined in a single-source multicasting application. This may further improve the multicasting efficiency and is discussed in Section 2.2 and 2.3. The increasing demand for multicasting applications leads to multi-source multicasting scenarios. The literature work related to multi-source multicasting in wireless networks is discussed in Section 2.4. Here packets of different multicasting sessions may be combined, using *inter-session* NC, for better overall throughput and delay performance for all the multicasting sessions in the network. Section 2.5 concludes the chapter.

## 2.1 Wireless Networks with Correlated Links

Opportunistic Routing (OR) [1] exhibits wireless broadcast advantage and improves the network performance. In OR, instead of pre-selecting a specific node as a next-hop forwarder for a packet, multiple nodes, usually called a Candidate Set, can potentially be selected as the next-hop forwarder. Hence, each node in OR can use different potential paths to send packets toward the destination. This is different from the traditional unipath routing which selects one next-hop forwarder before starting the transmission. By using OR, for each packet a dynamic route toward the destination is built, this is done according to the condition of the wireless links at the moment when the packet is being transmitted. Most of the existing protocols assume the signal paths (wireless links) from source to the neighboring nodes are independent of each other. The authors in [8] carried out empirical measurements using 802.11 indoor test-beds and discovered that the wireless links are not entirely independent of each other, i.e. a finite correlation may exist between them. A candidate set selection algorithm is presented in [8] for unicast transmissions considering wireless links that are correlated with each other. The authors in [9] have introduced a new kappa-metric. It captures the link correlation and determines the extent to which the links are correlated with each other. This metric provides a better understanding of the protocols which exploit wireless broadcast under a correlated links environment.

The impact of link correlations in wireless sensor networks is shown in [10] – [13] where an efficient data transmission technique under a correlated link environment is discussed. In [10], the nodes with smaller ETXs cost to the downstream nodes are selected as the core nodes that are used for data dissemination. This reduces the total transmission count and improves network performance when the wireless links are correlated with each other. Subsequently, in [11], authors exploit NC for data transmission where the network-coded

packet is formed based on feedback messages from the neighboring nodes. Collective Flooding (CF) technique may also be used in a wireless sensor network for transmitting data [12]. Here, the authors have proposed a data transmission protocol considering link correlations that reduces energy consumption and increases the lifetime of the wireless sensor nodes. A correlation aware OR protocol is proposed in [13], which combines the impact of link correlations and opportunistic forwarding to improve the network performance in lossy wireless sensor networks.

A general framework to capture correlations between the links in the wireless networks is proposed in [14]. The structure uses the signal to interference plus noise ratio (SINR) to accurately model the link correlations. The proposed model can be used to design the new NC and OR protocols in a wireless correlated link environment. The authors in [3] have shown the effect of link correlation in OR and proposed a novel metric that determines the average number of transmissions needed to unicast a packet from source to the destination in multi-hop wireless networks. A new forwarder set selection algorithm is proposed, which fully utilizes the wireless broadcast advantage with wireless correlated links. The work is evaluated with test-bed implementations and extensive simulations that show the impact of link correlation in network routing. A link correlation detection model for low-power wireless networks is proposed in [15] taking network layer parameters (packet length and packet transmission interval) into consideration. The proposed model is validated by incorporating it with the existing protocols for measuring link correlations. Experimental outcomes help to accurately measure the correlation between the links, which in turn results in better protocol performance.

A link-correlation aware cost metric that determines the expected number of packet transmissions in the network for unicast and broadcast communication is proposed in [16]. Wireless broadcasting is advantageous (because of diversity benefits) because the packet lost in one node may be received by another neighboring node that can then forward the packet further in the network.

Highly correlated neighboring links provide less diversity benefit as the probability of receiving and missing the packet at the neighboring nodes in the same transmission is high [16]. Random Linear Network Coding (RLNC) is an efficient NC approach that enables network nodes to generate independently and randomly linear mappings of input to output data symbols over a finite field. The authors in [16] use RLNC for broadcasting packets and discovered that the coding advantage is less when the wireless links are highly correlated. This is because the numbers of disjoint packets missing at the neighboring nodes that can be combined into one network-coded packet are less and therefore the coding gain gets reduced. An enhanced version of the *ETX* metric for wireless correlated links called cETX is proposed in [17]. The cETX metric provides forwarder set which has least cost from a source to the destination considering wireless correlation between the links. The authors have experimentally validated the effectiveness of the proposed metric in existing unicasting and broadcasting protocols. In [18], authors exploit OR to transmit packets in batches under the wireless correlated link environment where the destination node is present at a two-hop distance from the source. The source node communicates packets of a batch to its first-hop forwarding nodes. A forwarding node having the least *ETX* cost to the destination is called the best (highest priority) forwarder and communicates its received packets first. The other least priority forwarding nodes transmit the remaining packets according to their *ETX* cost to the destination node. In their second routing algorithm, the authors have incorporated OR

with NC in which the source node employs RLNC and combines the packets of a batch to form a network-coded packet and sends it to its forwarding nodes. The forwarding nodes randomly mix the packets present in their buffer and communicate them to the destination node.

The effect of link correlation in interconnecting Internet of Things (IoT) devices over multi-hop wireless networks is discussed in [19]. In a wireless network, the protocol performance is highly dependent on the link quality and the correlations between the neighboring links. In [20], each network node sends beacon messages at regular intervals to its corresponding neighboring nodes. The neighboring nodes record the reception of the messages in the form of bitmaps where '1' denotes packet reception, and '0' denotes packet loss. After receiving the bitmap information, the sender node updates its link quality with the neighboring nodes and the correlation between the neighboring links. The authors have also discussed a spoofing attack where a malicious node alters bitmaps information to force undue errors in link correlation measurements. This issue is addressed by proposing a beacon-trap approach [20] that identifies malicious nodes and avoids their impact on the protocol performance.

## **2.2 Reliable Multicast in Wireless Networks**

OR [1] explicitly exploits the inherent broadcast capability of wireless networks to improve on traditional routing techniques which are limited to using fixed intermediate nodes between the source and its destination. Methods for determining the appropriate relay nodes known as the Candidate Relay Set (CRS) for this routing are found for multi-casting in wireless sensor networks. The OR protocol was originally proposed to deal with the unreliability and time-varying nature of wireless link quality. OR is reported to be able to improve the overall performance of wireless networks with fewer transmissions, higher throughput and lower

latency. By not restricting to unicast communications that only focus on a single link, OR can utilize the possibility that more distant nodes can successfully receive a packet by “chance” and hence the term “opportunistic”. The Least Cost Anypath Routing (LCAR) algorithm introduced in [2] is a generalization of OR, which facilitates finding the candidate set with the least cost to the destination. In [21], OR is extended for multicasting scenarios (MC-OR) where packets are reliably transmitted in batches to all the destinations. MLCAR [5] is an extension of LCAR for multicast routing which exploits the opportunistic advantage of packet reception for selecting forwarding nodes which improve the multicasting efficiency. In MLCAR, multicasting costs to all possible subsets of the destination set are calculated at the nodes of the network, using a distributed Bellman-Ford type algorithm. The multicasting costs and the selected forwarding nodes are then used to route packets efficiently from the source to the intended destination subset. Efficient and scalable multicast routing schemes have also been proposed for Data Center Networks (DCN’s) in [22-25]. Since multicasting can support group communications, it may significantly reduce network traffic and improve the throughput of the DCN’s. In emergency applications like earthquake early warning systems, tsunami or tornado watch systems, the critical requirement is to efficiently disseminate the information to the possibly affected users as quickly as possible. In [26], the authors propose a multicasting protocol to deliver such critical information reliably to a large number of receivers. The Expected Multicast Transmission Count (EMTX) metric proposed in [27] evaluates the performance of such reliable multicast in multi-hop wireless networks. The design of the EMTX based multicasting tree takes into account the wireless broadcast feature and the link quality of the wireless links to reduce packet retransmissions and improve the overall performance of the system. The objective here is to minimize the sum of the EMTX between the forwarding nodes of the multicasting tree. Therefore, to increase the

network throughput, the authors have proposed a mathematical model to determine the multicasting tree whose overall EMTX value is the lowest.

Multicasting applications like multi-player gaming and text-based chats require multiple nodes to form a group, where each node of a group may communicate with all the other nodes present in the group. The communication between the nodes in a group can be performed by using a relay node, and therefore the network is known as multi-group multi-way relay networks [28]. Here, multiple antennas mounted at each relay node are divided into clusters where each cluster of antennas serves a specific number of nodes in a group. Using the algorithm proposed in [28], one can increase the number of nodes (users) served by the same number of relay antennas, thereby saving the antenna resources which are limited in general. Further, to increase the multicasting throughput, authors in [29] have used the beamforming approach to multicast data from a single transmitter to a large number of receivers within a group. Device-to-Device (D2D) multicast communication is also becoming increasingly important for applications in densely populated areas such as supermarkets or a stadium where a large number of users are likely to share the same type of contents. In a typical D2D multicasting scenario, the users are allocated in different disjoint clusters, and for each cluster, a particular user is selected to be the cluster head (CH). The CH is responsible for communicating the entire content that needs to be shared between all the users in a cluster by using D2D multicasting. The CHs may suffer from mutual interference which is mitigated by using a game-theoretic approach in [30]. The performance analysis of D2D multicasting in wireless networks is presented in [31].

NC is a powerful technique to take into account the information content of each transmission and use that knowledge to improve network throughput further. The MORE [4] is a reliable multi-hop multicast routing protocol where packets in a batch are combined using

RLNC and transmitted to the forwarding nodes. All the forwarding nodes linearly combine the packets stored in their respective buffers to form a coded packet for further downstream transmission. Once the destination collects a sufficient number of coded packets to recover the original packets in a batch, it sends back an ACK to the source to trigger the transmission of a new batch, if any. Therefore, here, the network performance is governed by the weakest link from the source to destinations. In any random network, performing NC in every network node would be uneconomical. To reduce costs, NC is limited to a subset of the network nodes, and the necessary and sufficient condition for a node to act as an encoding node is proposed in [32]. Note that this approach would reduce the network throughput by limiting the number of nodes where NC may be done. Pacifier [7] addresses the shortcoming of MORE by allowing the source to transmit the next batch, every time the batch is received by one of the receivers, thereby improving the throughput of the well-connected destinations. After, the source has finished the transmission of the last batch; it retransmits the batches which have not yet been received at all the destinations, and does this starting from the first missing batch. Pacifier improves the time taken to transmit the batch to the well-connected destinations but increases the delays for the poorly linked ones [7]. In the process, the network may lose a large number of packets due to frequent batch switching. To overcome these issues, a NC Assisted MLCAR (NC-MLCAR) scheme is proposed in Chapter 4 where NC is performed after determining the *ETX* based least cost multicasting route. NC-MLCAR outperforms both MORE and Pacifier by transmitting a new batch every time the desired downstream nodes successfully receive the current batch.

Wireless relay assisted multicasting scheme using binary NC with a single source, single relay, and two destinations is presented in [33]. The source node broadcasts packets for a fixed time period and collects the packet receptions status from the relay and destinations

using ACKs and NACKs. Packets received at the relay are combined and then reliably communicated to destinations. The source subsequently retransmits the packets which are missing at the destinations. Wireless network performance often degrades due to the interference caused by subsequent transmissions at the neighboring nodes. In order to improve the performance of the network, the authors in [34] and [35] have proposed scheduling techniques which carefully perform interference-free multicast transmission using RLNC over multi-hop wireless networks. In [36], the authors consider random arrival of packets at the source node and the queuing delay associated with them. A study of queuing delay analysis is performed for multicasting using RLNC over a one-hop wireless network. NC provides throughput gain at the cost of decoding delay as all the multicasting destinations need the required number of packets to decode the information successfully. The authors have proposed a scheduling scheme in [37] which determines the packets that have to be mixed together to form a coded packet and the intermediate nodes of the network, which will perform this coding operation. The proposed coding-aware scheduling scheme reduces the network delay and bandwidth requirement and improves the transmission efficiency of the network. Similarly, in order to reduce the decoding delay at the end-users, scheduling algorithms are discussed in [38] and [39] for a network where the source node communicates a data block of  $K$  packets using an intermediate node, to multiple users which are two-hops away from the source. The intermediate node stores the packets successfully received from the source in its buffer, and forms a coded packet which is then transmitted to all the multicasting destinations. With the help of the proposed scheduling algorithm, an intermediate node decides either to fetch a new packet from the source or to multicast a network-coded packet to the receivers, using the packets already available in its queue.

With increasing demand for wireless applications, many schemes involving multi-rate multi-channel communications have been proposed to improve the efficiency of wireless networks. Multicasting using intra-session NC in a multi-rate, multi-channel scenario is discussed in [40], where the authors have proposed an optimization model to improve the throughput of the network at the cost of increased network resource usage. In [41], an optimization strategy has been suggested for reliably multicasting data to one-hop destinations using RLNC. Recently, in [42], the authors have analyzed its performance in terms of probability of successful packet delivery to an arbitrary number of users. The authors in [43] employ optimal decoders at the receivers, which are one hop away from the source. The proposed binary NC scheme combines the disjoint packets missing at the receivers. The coded packet is then broadcast to the desired receivers for better multicasting efficiency. A topology-aware NC technique is proposed in [44] to reduce the expected number of packet transmissions for multicasting data to one-hop destinations. With the help of network topology information, different lost packets are combined using RLNC into one coded packet, which is then broadcast to the desired users. A binary NC scheme is proposed in [45] for multicasting to one-hop users. For multi-hop wireless multicasting, the authors in [46] propose an energy-efficient algorithm where directional antennas are used for communicating network-coded packets to the destination nodes. The intermediate nodes stay in a sleeping mode when not in use, in order to conserve power. The combined impact of NC and the sleeping scheme keep energy consumption low while improving multicasting transmission efficiency.

### 2.3 Unreliable Multicast in Wireless Networks

Multicasting in real-time multimedia applications like video-on-demand, voice-over-IP, internet protocol television, and video-conferencing has hard deadlines on packet transmission while maintaining a high quality-of-service (QoS). This is achieved at the cost of some packet loss. The increasing demand for multimedia applications in mobile ad-hoc networks (MANETs) is addressed by the authors in [47]. An algorithm is proposed to construct a bandwidth-satisfied multicasting tree connecting a source to its corresponding destinations where forwarders participating in packet transmission are selected to reduce the overall bandwidth and power consumption for the network. Since most real-time and multimedia applications are delay-sensitive, the design of multicast routing protocol considers end-to-end delay. A multicast routing protocol is presented in [48] that exhibits acceptable packet transmission and delivery while satisfying the required average bandwidth and end-to-end delay requirements in MANETs. The authors in [49]-[51] have proposed delay-constrained multicast routing algorithms that determine a path that delivers data to all the destination users within a specific upper-bounded end-to-end delay time while minimizing the energy consumption over the wireless network.

Wireless video-streaming applications like live football matches or music concerts require delivering a good video quality within a predefined threshold time. Therefore, a Scalable Video Transmission (SVT) technique that divides the video stream into multiple layers is generally used to multicast [52]-[54]. The most significant layer is the base layer (BL) which contains the essential features of a video and therefore ensures the necessary video quality.

The additional layers known as enhancement layers (ELs) are used for further quality refining. The viewing experience (video quality) at the user end improves with an increase in the number of ELs. Therefore, the primary objective of SVT is that the users should successfully receive all the packets belonging to the BL and a maximum number of packets from the ELs within a specific upper-bounded delay. Further, a downlink scheduler is proposed in [53] that provides multicasting gain to the users present over one-hop distance from the source; the users may arrive and leave the network in between the running transmissions. The optimization techniques to improve efficiency on video multicasting to clients in a two-hop network with a designated relay node are discussed in [54] and [55]. In [56], the authors have proposed an algorithm for multicasting in the cellular networks which ensures that all the mobile users in a cell receive satisfactory video quality.

NC is used to further improve the multicasting efficiency through scalable video transmissions which are popularly known as Scalable Video Coding (SVC) [57]. In SVC, the BL is the essential layer; therefore the packets of the BL should be protected during transmission. Once the BL is successfully reconstructed at the user end, packets of the ELs can be transmitted in a hierarchical manner to improve the video quality. Here, the second EL is useful only when the first EL is successfully received at the user end. Thus, multicast video-stream in SVC involves broadcasting packets of the BL followed by that from multiple ELs to multiple clients present over one-hop distance from the Base Station (BS). Due to the lossy nature of wireless networks, some clients may not receive a sufficient number of packets in order to retrieve a good quality video. In [58], an efficient packet retransmission

strategy is presented that imposes a hard deadline constraint on data transmission. The BS uses the RLNC technique to combine the missing packets such that the clients are able to get the desired video quality with fewer transmissions. In [59], a packet-level scheduling technique with a predefined threshold time is proposed to multicast data. The authors in [60] have proposed an optimization framework that reduces the complexity of the RLNC decoder and ensures the desired video quality at all the multicasting clients.

Due to the heterogeneous nature of the wireless links, multicast network performance is limited by the poorest link connecting source to the destination [61]. Using an omnidirectional antenna may result in the wastage of power and bandwidth in non-targeted areas where no users are present. In order to address these issues, a physical layer beamforming technique is incorporated with packet scheduling. This improves the performance of all the users present at one-hop distance from the BS [61]. The joint use of beamforming and RLNC for packet transmission maximizes the overall throughput of the network. Recently, in [62], Software-Defined Networking (SDN) is used for multicasting a video-stream over a multi-hop wireless network. SDN consists of a central controller that checks network traffic and manages forwarding elements like switches and routers. The controller identifies the least congested path to the BL in order to provide the desired video quality to all the users. The controller collects the information on network topology and available end-to-end bandwidth, which can be used to build an efficient multicasting tree that, maximizes the overall video quality.

Wireless multicasting of the multimedia applications requires maximum throughput with minimum delay. A scheduling algorithm is proposed in [63] where users are one-hop away from the source (e.g., Wi-Fi networks). The source node exploits RLNC to intelligently

combine the native packets to reduce the packet delay to reach the receiving end. A multicasting delay-sensitive application over a multi-hop wireless network is discussed in [64]. The multicasting data is divided into blocks of packets, and the packets belonging to each block should be successfully received at the receiver's end within a specific threshold time. An intra-session RLNC is applied where packets belonging to the same block are combined and transmitted to the forwarding nodes. A forwarding node combines the packets present in its buffer using RLNC and communicates it further. To improve the multicasting efficiency, packet scheduling is performed such that the destinations receive the maximum number of packets of a block within a predefined threshold time.

The multicasting efficiency of the wireless networks generally depends on being able to reach the more weakly connected destinations from the source. This can be tackled by using multi-rate multicasting, which allows transmitting at different transmission rates to different users leading to better network resource utilization. This kind of multi-rate multicasting to multiple clients which are more than two-hops away from the BS, using multiple relay nodes, is presented in [65]. Block-wise packet transmission is done where BS broadcasts a block of  $N$  native packets such that all the relay nodes receive the block. The clients may receive some native packets, and the missing packets are communicated from the relay nodes using RLNC. The relay nodes transmit network-coded packets using different transmission rates such that the overall packet transmission delay is minimized. In [66], authors have presented multi-rate multicasting to improve the throughput of the multi-hop wireless network. However, authors in [67] have used multiple channels with multiple radios at every network node to improve the multicasting efficiency in multi-hop wireless networks. Simulation-based performance evaluation is presented considering end-to-end packet delay, packet delivery ratio, and file completion time.

Wireless communications tend to be error-prone since the performance of the network generally depends on multiple factors such as fading, interference, mobility, and quality of the wireless channel. Multicasting routing protocols should be designed for optimum network resource utilization. Bandwidth-satisfied multicasting protocols for MANETs are proposed in [68] and [69]. The multicasting tree is designed to reduce the overall bandwidth consumption of the network. Here, the nodes use network-coding in the tree, which further improves the multicasting efficiency. In [70], an OR protocol enhancing MORE [4] for video-streaming applications is proposed where the destination nodes are at most two hops away from the source. In a typical wireless network where users are one hop away from the BS, it is possible that a user may get all the desired network-coded packets and can then act as a secondary source for the other users (D2D communication). The authors in [71] have proposed a hybrid multicast and D2D transmission scheme where source node transmits data in batches of packets. In every time-slot, BS decides to operate in either the multicast mode or the D2D mode. In the case of multicasting mode, the BS scheduler combines packets of a batch using RLNC and broadcasts it to all the users. When the D2D method is selected, one of the users is chosen as the source to transmit a network-coded packet to the other users. This scheduling scheme combines the impact of multicasting and D2D communication to improve the overall throughput of the network.

#### **2.4 Multi-Source Multicast in Wireless Networks**

Multi-source multicasting caters to emerging applications where different clients demand data from their respective source nodes as in multimedia applications, software updates, billing records, electronic newspapers, and medical images. In [72], authors introduce multi-source multicasting where a sub-network is designated for each source. The primary aim is to satisfy the end-to-end delay bound of all the clients demanding multimedia applications. The

authors in [73] further strengthen multimedia streaming applications consisting of multiple multicasting sessions. One of the critical aspects of wireless transmissions is link scheduling, as discussed in [74] where various source nodes transmit data to their respective users located at one-hop distance from the source nodes. The main concern is to reduce interference at the destination nodes due to simultaneous transmissions as this may improve the overall network throughput. An efficient multi-flow multicasting algorithm is proposed in [75] for multimedia applications. The flow scheduling policy admits new multicasting flow without affecting the existing scheduled transmissions. This scheme improves the capability of the networks to acknowledge new multicasting flows to utilize the network resources efficiently. Meanwhile, authors in [76] introduce multi-source multicasting for information-centric networks. A spanning tree is built for sending data to the different multicasting groups. In [77], the authors discuss multi-source video multicasting scenarios over a wireless mesh network. They have introduced a resource-efficient algorithm employing efficient bandwidth utilization and performance-guaranteed scheduling for all the multicasting sessions, where different sources can share an integrated routing path to control overhead costs due to individual routes.

Wireless broadcasting gives an opportunity for performing NC, where a single coded packet is beneficial for different sessions. COPE is introduced in [78] for any random topology where an inter-session NC is used for transmitting packets of an arbitrary number of unicast flows that arrive and leave dynamically. The authors of [79] aims to reduce the coding complexity for a multi-source multi-destination network by eliminating the encoding nodes. An optimal path that does not require a coding node is chosen considering all the sources and their corresponding destinations. In [80], an analytical expression is obtained to show the impact of random NC on the system delay in a multi-source, multi-destination scenario. This work explicitly focuses on two-hop transmissions, where relay nodes are used

to improve network throughput. Subsequently, in [81], packet-wise data transmission of multiple unicasting sessions using inter-session NC with two common destination nodes is studied. With the help of the packet reception reports from the destination nodes, the relay node broadcasts a new inter-session coded packet to improve the overall system performance. In [82], the authors study the multi-source, multi-destination case for two-hop wireless networks having a relay node. This work is based on batch-wise transmission where each source broadcasts packets in batches. The destination nodes send packet reception status to the relay node using which the node transmits the missing packets on performing an inter-session NC. Following the batch-wise transmission, the authors in [83], [84] show advantage of an intra-session and inter-session NC in case of multiple unicasting sessions over multi-hop wireless networks. An intra-session NC increases redundancy, and inter-session coding improves the overall network throughput. In [84], the authors have mainly focused on grouping packets of different sessions to make an efficient use of inter-session NC.

Furthermore, using intra-session and inter-session NC schemes, the authors in [85] transmit data over a two-hop relay network with two unicasting sessions. It is found that increasing the number of sessions adds to the coding advantage because more packets are available for NC. For applications like video streaming, authors in [86] use inter-session NC to transmit packets of two unicasting sessions with hard deadline constraints. Thus, the packets of each unicasting session must reach the corresponding destination set within a specific threshold interval; otherwise the packets are considered to be useless. This work suggests packet transmissions in small batch sizes to maximize the overall number of successful packet transmissions. Applications like search and rescue operations, satellite communication, and Wi-Fi communication involve broadcasting of information from the source node to all the neighboring nodes. The authors in [87] discuss a multi-source broadcasting strategy where

packets in the batches formed at different sources are combined and transmitted to increase the packet delivery ratio of the network.

NC for multi-source multicasting scenarios is considered in [89], where the original network is divided into several sub-graphs with a source and its corresponding destination set. Linear NC is performed at each source having a different destination set. The routing paths of different multicasting sessions do not coincide with each other. Routing and packet scheduling algorithms for multicast sessions where network nodes apply intra-session RLNC to form a network-coded packet are proposed in [90]. The authors in [91] provide support to the sources of different multicasting sessions which are unable to meet flow-constraints at the cost of reduced throughput. NC scheme for multi-source multicasting, which improves the throughput of the network using intra-session and inter-session NC is proposed in [92]. The transmission topology (TT) problem for multi-source multicasting is approached using a game-theoretic method and the computational complexity is reduced by employing NC in [93]. Multimedia multicasting with a continuous demand for the same content from various sources to the same set of destinations is considered in [94]. This work provides an optimal transmission topology (TT) that employs a secure linear NC (SLNC) technique to protect the information from an eavesdropper. The authors of [95] address security concerns and introduce RLNC based secure multi-source multicasting for real-time streaming applications where the data transmission is initiated in the presence of an eavesdropper.

## **2.5 Summary**

Wireless broadcasting advantage depends on the correlation between the wireless links. The correlation between the wireless links allows a better understanding of wireless routing protocols. Multicasting supports one-to-many communications, and its applications are

characterized based on the requirement for either reliable or unreliable data transmissions. Reliable multicasting requires strict packet reliability at the receiver's end as in software updates, electronics newspapers, and medical images. On the other hand, unreliable multicasting deals with hard deadline constraint-based data transmissions like video streaming applications and on-demand video services. This chapter discussed both reliable and unreliable multicasting schemes in detail when the source node and its corresponding destination nodes are present anywhere in the network. NC assists in the wireless broadcast by combining missing packets into a coded packet that is then subsequently transmitted to the corresponding receivers. This chapter discussed the impact of NC on both reliable and unreliable multicast transmissions. A multi-source multicasting protocol design is also a relevant research domain in this context. The presence of packets from different sessions at a network node provides an opportunity to perform both intra-session and inter-session NC. Intra-session NC increases the multicasting efficiency of a session, and inter-session NC improves the overall throughput of the network.





## Chapter 3

# Multicasting in Wireless Networks with Correlated Links

This chapter presents our approach for cost calculations that may be followed for multi-hop multicasting over correlated wireless links. The MLCAR scheme of [5] is used to identify the least cost route for networks with known link correlations, and the results are compared to a network with no information on link correlations. Analytical results obtained for a multicast network with correlated links are subsequently verified through simulations.

### 3.1 Introduction

Multicasting in wireless networks involves simultaneous transmission of information to multiple receivers. The links connecting the transmitting and the receiving devices may not be completely independent of each other. In that case, the number of transmissions needed to send a packet reliably from the source to the corresponding destinations would differ from that for a network where the links are independent, even if the individual link costs are the same in both cases. The cost metrics for correlated links are computed using the direct link costs and the correlation between the links. In MLCAR, a Bellman-Ford type approach is used to calculate the multicasting cost from each source to all possible subsets of its destination set. Based on this cost, the forwarder set is selected and the strategy is decided upon to transmit the packet to all nodes in the destination set with the lowest multicasting cost. It may be noted that when link correlations are taken into account, the least-cost route to the destination subset may change in some cases. In [1] and [2], the authors assumed the simple scenario where all the links of the wireless network are independent. However, in practical wireless implementations, there may be a finite correlation between different

(typically neighboring) links as in [3] and [8] - [20]. The links of a network are considered to be highly correlated if the receiving nodes have higher chances of successfully receiving the packet in the same transmission. When links have low correlation, the probability of the packet traversing the links correctly in the same transmission would be low.

The rest of the chapter is organized as follows. In Section 3.2, an example is presented to illustrate our approach to calculate the multicasting cost in a correlated wireless link environment. Section 3.3 presents the multicast route cost calculations with correlated links for a generalized network. In Section 3.4, a packet transmission strategy is introduced. In Section 3.5, the simulation results are discussed and finally the chapter is concluded in Section 3.6.

### 3.2 Multicasting over a Network with Correlated Links

Consider a simple one-hop wireless multicasting network with a source  $S^{(l)}$  and two destinations  $D_1$  and  $D_2$  as shown in Fig. 3.1. The link cost metric is given as  $ETX_{i,j} = 1/p_{i,j}$ . The multicasting cost to send a packet to both destinations using a simple link-by-link unicasting approach will be 4.50 (2.00+2.50). When wireless broadcasting is being used and node  $S^{(l)}$  broadcasts a packet, it is possible that the packet is received by either or both of the two destination nodes  $D_1$  or  $D_2$ . If the packet does not reach either of the destinations  $D_1$  or  $D_2$ , then node  $S^{(l)}$  has to retransmit the packet. The probability of correlation between the links  $S^{(l)}-D_1$  and  $S^{(l)}-D_2$  are considered to be  $P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2})$ , where  $E_{S^{(l)},D_i}$  is the event (probability) that the packet transmitted at  $S^{(l)}$  is successfully received by node  $D_i$ ,  $i \in (1, 2)$ . For a packet transmitted by node  $S^{(l)}$ , the following events may happen

1. Packet is not received at both  $D_1$  and  $D_2$ .
2. Packet is received at  $D_1$  but not at  $D_2$ .
3. Packet is received at  $D_2$  but not at  $D_1$ .
4. Packet is successfully received at both  $D_1$  and  $D_2$ .

In cases 1 to 3, the packet is retransmitted by  $S^{(l)}$  to the intended destination(s). The multicasting cost of the network in Fig. 3.1 is then given by-

$C_{S^{(l)},\{D_1,D_2\}}^{(l),w}$  = (probability that both  $D_1$  and  $D_2$  do not receive the packet)  $\times$  (1 + cost of multicasting the packet at  $S^{(l)}$ ) + (probability that  $D_1$  receives the packet and  $D_2$  does not)  $\times$  (1 + cost of transmitting the packet from  $S^{(l)}$  to  $D_2$ ) + (probability that  $D_2$  receives the packet and  $D_1$  does not)  $\times$  (1 + cost of transmitting the packet from  $S^{(l)}$  to  $D_1$ ) + (probability that both  $D_1$  and  $D_2$  receive the packet successfully).

This multicasting cost is examined under two scenarios.

*Scenario 1: Links are independent*

For the network in Fig. 3.1

- probability that both  $D_1$  and  $D_2$  do not receive the packet =  $(1 - p_{S^{(l)},D_1})(1 - p_{S^{(l)},D_2}) = 0.30$
- probability that  $D_1$  receives the packet and  $D_2$  does not =  $(p_{S^{(l)},D_1})(1 - p_{S^{(l)},D_2}) = 0.30$
- probability that  $D_2$  receives the packet and  $D_1$  does not =  $(1 - p_{S^{(l)},D_1})(p_{S^{(l)},D_2}) = 0.20$
- probability that both  $D_1$  and  $D_2$  receive the packet successfully =  $(p_{S^{(l)},D_1})(p_{S^{(l)},D_2}) = 0.20$

On substituting, the multicasting cost in case of independent link scenario is given by-

$$C_{S^{(l)},\{D_1,D_2\}}^{(l),i} = (0.30) \times (1 + C_{S^{(l)},\{D_1,D_2\}}^{(l),i}) + (0.30) \times (1 + 2.50) + (0.20) \times (1 + 2) + (0.20).$$

On simplifying,  $C_{S^{(l)},\{D_1,D_2\}}^{(l),i} = 3.07$ .

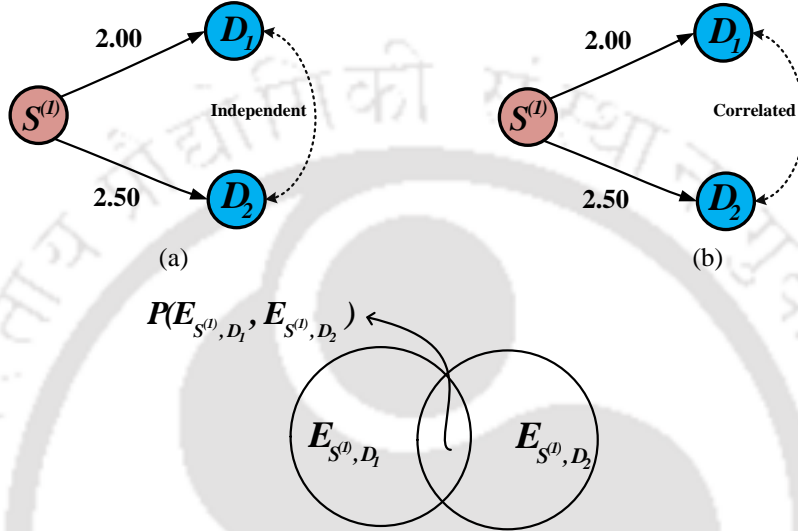


Figure 3.1: Example network 1 with correlation between the links

Scenario 2: Links are correlated with probability of correlation,  $P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2}) = 0.38$

In this case

- probability that both  $D_1$  and  $D_2$  do not receive the packet

$$= 1 - p_{S^{(l)},D_1} - p_{S^{(l)},D_2} + P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2}) = 0.48$$

- probability that  $D_1$  receives the packet and  $D_2$  does not

$$= p_{S^{(l)},D_1} - P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2}) = 0.12$$

- probability that  $D_2$  receives the packet and  $D_1$  does not

$$= p_{S^{(l)},D_2} - P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2}) = 0.02$$

- probability that both  $D_1$  and  $D_2$  receive the packet successfully

$$= P(E_{S^{(l)},D_1}, E_{S^{(l)},D_2}) = 0.38$$

On substituting, the multicasting cost in case of correlated link scenario is given by-

$$C_{S^{(l)},\{D_1,D_2\}}^{(l),c} = (0.48) \times (1 + C_{S^{(l)},\{D_1,D_2\}}^{(l),c}) + (0.12) \times (1 + 2.50) + (0.02) \times (1 + 2) + (0.38).$$

On simplifying,  $C_{S^{(l)},\{D_1,D_2\}}^{(l),c} = 2.58$ .

It has been observed that the multicasting costs differ for both the scenarios. This shows the importance of considering link correlation while calculating multicasting cost. In Scenario 1, the multicasting cost is computed as 3.07. However, if the links are considered correlated (scenario 2), the difference in the multicasting cost over the network with independent links is around 16%. This shows that the link correlation has a significant impact on the multicasting cost.

### 3.3 Multicasting Route Cost Calculation with Correlated Links

Consider a multicast network where source node  $S^{(l)}$  broadcasts packets over a wireless medium intended for destination set  $DS^{(l)} \{D_1, \dots, D_{M^{(l)}}\}$ .  $S^{(l)}$  is connected to  $DS^{(l)}$  through first-hop neighbors  $NS_n$ . A forwarder set  $f_x^{(l),c}$  is selected from the first-hop neighbors of  $S^{(l)}$ , such that, it reaches all nodes in  $DS^{(l)}$ . Nodes in  $f_x^{(l),c}$  act as  $S^{(l)}$  for the second-hop neighbors and the forwarder set for the selected  $S^{(l)}$  are its first-hop neighbors. At each successive hop,  $S^{(l)}$  and  $f_x^{(l),c}$  are updated until all the destinations are covered. The link cost and the probability of correlation between the links are assumed to be known. The link cost metric of link from  $i$  to  $j$  is  $ETX_{ij} = 1/p_{i,j}$ , where  $i \neq j$ ,  $j$  is first-hop downstream neighbor of  $i$ , and  $p_{i,j}$  is the probability of successful packet transmission over the link  $i - j$ .

The probability of correlation between  $L$  links is defined as the probability of the event when the packet is successfully received at all the  $L$  nodes simultaneously, and is represented as  $P(E_{S^{(l)},f_1}, \dots, E_{S^{(l)},f_L})$ , where  $E_{S^{(l)},f_i}$  is the event when the packet transmitted by  $S^{(l)}$  is successfully received at  $f_i$ ,  $i \in (1, \dots, L)$ . The multicast route cost is the ETXs needed for reliably delivering a packet from  $S^{(l)}$  to  $DS^{(l)}$  using forwarder set  $f_x^{(l),c}$ . At every successive hop the transmitting node is considered as  $S^{(l)}$ , the direct intended receivers which reach at least one node of  $DS^{(l)}$  belongs to the forwarder set  $f_x^{(l),c}$ . Let  $f_a^{(l),c}$  be the subset of  $f_x^{(l),c}$  which receives the packet with probability  $\beta_{f_a^{(l),c}}^{(l),c}$  and  $f_y^{(l),c}$  be the subset of  $f_x^{(l),c}$  which does not receives the packet. In case of network with correlated links, the multicast route cost for transmitting a packet from  $S^{(l)}$  to  $DS^{(l)}$  using  $f_x^{(l),c}$  is given by-

$$C_{S^{(l)}, DS^{(l)}}^{(l),c}(f_x^{(l),c}) = \frac{1 + \sum_{\forall f_a^{(l),c} \subseteq f_x^{(l),c}, f_a^{(l),c} \neq \emptyset} \beta_{f_a^{(l),c}}^{(l),c} \left( C_{f_a^{(l),c}, DS^{(l)}}^{(l),c} + C_{S^{(l)}, f_y^{(l),c}}^{(l),c} \right)}{\sum_{k=1}^{|f_x^{(l),c}|} (-1)^{k-1} P(f^k)} \quad (3.1)$$

Here  $f^k$  is subset of  $f_x^{(l),c}$  with size  $k$  and  $P(f^k)$  is probability that  $k$  nodes of  $f_x^{(l),c}$  successfully receives the packet and communicate ACK to the source.  $P(f^k)$  in (3.1) is written as-

$$P(f^k) = \sum_{f^k \subseteq f_x^{(l),c}} P(E_{S^{(l)},f_1^k}, E_{S^{(l)},f_2^k}, \dots, E_{S^{(l)},f_k^k}) \quad (3.2)$$

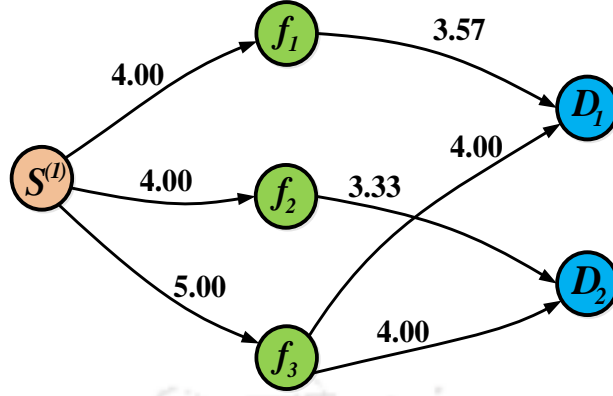


Figure 3.2: Example network 2: Multicasting

The multicast route cost for an upstream node  $u$ , which is directly connected to the destination  $d$  ( $d \in DS^{(l)}$ ) is the same as the  $ETX$  cost, that is  $C_{u,d}^{(l),c} = ETX_{u,d}$ . Cost calculation for the example network in Fig. 3.2, where  $S^{(l)}$  multicasts a packet to  $DS^{(l)}$  ( $D_1, D_2$ ) considering  $f_x^{(l),c} = \{f_1, f_2, f_3\}$  is shown. The other possible options for  $f_x^{(l),c}$  are  $\{f_3\}$ ,  $\{f_1, f_2\}$ ,  $\{f_1, f_3\}$ , and  $\{f_2, f_3\}$ . The  $ETX$  cost of each link is mentioned in Fig. 3.2. The probabilities of correlation between different links are assumed to be  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_2}] = 0.23$ ,  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_3}] = 0.02$ ,  $P[E_{S^{(l)},f_2}, E_{S^{(l)},f_3}] = 0.02$ ,  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_2}, E_{S^{(l)},f_3}] = 0.01$ ,  $P[E_{f_3,D_1}, E_{f_3,D_2}] = 0.24$ . For the example network in Fig. 3.2, the denominator of (3.1) can be written as-

$$\sum_{k=1}^{|f_x^{(l),c}|} (-1)^{k-1} P(f^k) = p_{S^{(l)},f_1} + p_{S^{(l)},f_2} + p_{S^{(l)},f_3} - P(E_{S^{(l)},f_1}, E_{S^{(l)},f_2}) - P(E_{S^{(l)},f_1}, E_{S^{(l)},f_3}) - P(E_{S^{(l)},f_2}, E_{S^{(l)},f_3}) + P(E_{S^{(l)},f_1}, E_{S^{(l)},f_2}, E_{S^{(l)},f_3}) \quad (3.3)$$

---

**Algorithm 1: Route cost calculation**  $(n, DS_n^{(1)}, NS_n, NN, C_{n,DS_n^{(1)}}^{(1,c)}, FS_n^{(1)})$ 


---

1.  $f_x^{(1,c)} = \{\phi\}, FS_n^{(1)} = \{\phi\}, NS_n = \{f_1, \dots, f_l\}, DS_n^{(1)} = \{D_1, \dots, D_{M^{(1)}}\}$
  2. **if**  $n \notin DS_n^{(1)}$  **then**
  3.  $C_{n,DS_n^{(1)}}^{(1,c)} = \infty$
  4. **else**  $C_{n,DS_n^{(1)}}^{(1,c)} = 0$ ;
  5. **for each**  $f_x^{(1,c)} \subseteq NS_n$  **do**
  6.  $C_{n,DS_n^{(1)}} = 0$
  7. **for each**  $f_a^{(1,c)} \subseteq f_x^{(1,c)}$  **do**
  8.  $C_{f_a^{(1,c)}, DS_{f_a^{(1,c)}}^{(1,c)}}^{(1,c)} = \infty$
  9. **for each**  $f_{aa}^{(1,c)} \subseteq f_a^{(1,c)}$  **do**
  10.  $C_{opt}^{(1,c)} = 0$  and  $DS_{f_a^{(1,c)}}^{(1,copy)} = DS_{f_a^{(1,c)}}^{(1)}$
  11. **for each**  $i \in f_{aa}^{(1,c)}$  **do**
  12. **for each**  $j \in DS_{f_a^{(1,c)}}^{(1,copy)}$  **do**
  13.  $C_{min} = \min C_{i,j}$  ;;
  14.  $C_{opt}^{(1,c)} = C_{opt}^{(1,c)} + C_{min}$  and  $DS_{f_a^{(1,c)}}^{(1,copy)} = DS_{f_a^{(1,c)}}^{(1,copy)} - j$
  15. **if**  $DS_{f_a^{(1,c)}}^{(1,copy)} \neq \{\phi\}$  **then**
  16.  $f_{aa}^{(1,c)} = f_{aa}^{(1,c)} - i$
  17. **if**  $f_{aa}^{(1,c)} \neq \{\phi\}$  **then**
  18. **go to** 11
  19. **else go to** 9 ;;
  20. **if**  $C_{opt}^{(1,c)} < C_{f_a^{(1,c)}, DS_{f_a^{(1,c)}}^{(1,c)}}^{(1,c)}$  **then**
  21.  $C_{f_a^{(1,c)}, DS_{f_a^{(1,c)}}^{(1,c)}}^{(1,c)} = C_{opt}^{(1,c)}$  ;;
  22.  $C_{n,DS_n^{(1)}} = C_{n,DS_n^{(1)}} + \beta_{f_a^{(1,c)}}^{(1,c)} (1 + C_{f_a^{(1,c)}, DS_{f_a^{(1,c)}}^{(1,c)}}^{(1,c)} + C_{n,DS_{f_y^{(1,c)}}^{(1,c)}}^{(1,c)})$  ;
  23. **if**  $C_{n,DS_n^{(1)}} < C_{n,DS_n^{(1)}}^{(1,c)}$  **then**
  24.  $C_{n,DS_n^{(1)}}^{(1,c)} = C_{n,DS_n^{(1)}}$  and  $FS_n^{(1)} = f_x^{(1,c)}$  ;;
- 

In Fig. 3.2, it is observed that  $D_1$  can be accessed using  $\{f_1, f_3\}$  and  $D_2$  using  $\{f_2, f_3\}$ . The cost of direct links from node  $f_i$  for  $i = \{1, 2, 3\}$  to destination nodes  $D_1$  and  $D_2$  is given in Fig. 3.2. The cost is infinity if the corresponding link between the forwarder node and the

destination node does not exist. The forwarding node  $f_1$  and  $f_2$  are connected to the subset of  $DS^{(l)}$  through a direct link, thus the costs from  $f_1$  to  $D_1$  and  $f_2$  to  $D_2$  is same as the link cost. The cost of a packet transmitted from  $f_3$  to be successfully received by  $D_1$  and  $D_2$  is computed using (3.1). The computed costs at  $f_1, f_2$ , and  $f_3$  are propagated to their upstream node  $S^{(l)}$ .

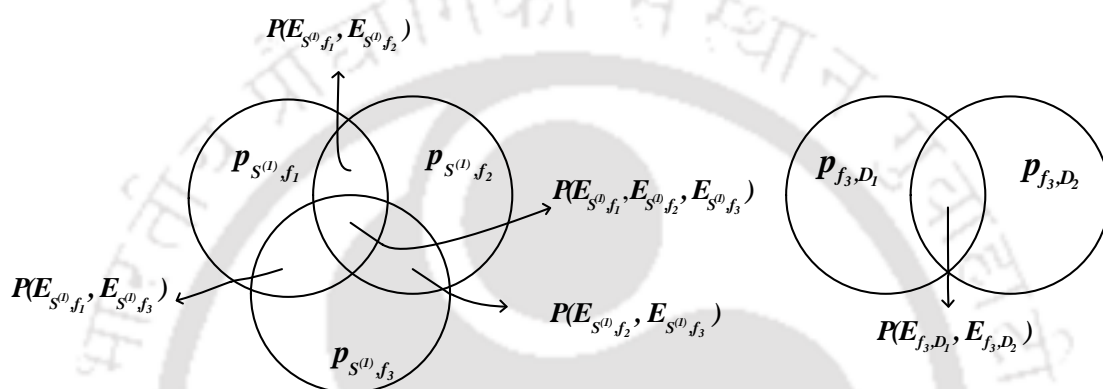


Figure 3.3: Venn diagram showing correlation between the links of the example network 2

For  $f_x^{(1),c} = \{f_1, f_2, f_3\}$ , if a packet communicated from  $S^{(l)}$  is received by node  $\{f_1\}$  alone, then  $f_a^{(1),c} = \{f_1\}$ ,  $f_y^{(1),c} = \{f_2, f_3\}$ ,  $DS_{f_a^{(1),c}}^{(1)} = \{D_1\}$ ,  $DS_{f_y^{(1),c}}^{(1)} = \{D_2\}$ , and  $\beta_{f_a^{(1),c}}^{(1),c} = p_{S^{(l)}, f_1} - P[E_{S^{(l)}, f_1}, E_{S^{(l)}, f_2}] - P[E_{S^{(l)}, f_1}, E_{S^{(l)}, f_3}] + P[E_{S^{(l)}, f_1}, E_{S^{(l)}, f_2}, E_{S^{(l)}, f_3}] = 0.01$ . In this case,  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 3.57$  from Fig. 3.2 and  $C_{S^{(l)}, DS_{f_y^{(1),c}}^{(1)}}^{(1),c} = 5.94$  using (3.1). Similarly, for  $f_a^{(1),c} = \{f_2\}$ ,  $DS_{f_a^{(1),c}}^{(1)} = \{D_2\}$ ,  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 3.33$ , and  $C_{S^{(l)}, DS_{f_y^{(1),c}}^{(1)}}^{(1),c} = 6.07$ .  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 4.15$  for  $f_a^{(1),c} = \{f_3\}$  and  $DS_{f_a^{(1),c}}^{(1)} = \{D_1, D_2\}$ . For  $f_a^{(1),c} = \{f_1, f_2\}$ ,  $f_y^{(1),c} = \{f_3\}$ ,  $DS_{f_a^{(1),c}}^{(1)} = \{D_1, D_2\}$ ,

$DS_{f_y^{(1),c}}^{(1)} = \{\phi\}$ , and  $\beta_{f_a^{(1),c}}^{(1),c} = P[E_{S^{(1)}_{f_1}}, E_{S^{(1)}_{f_2}}] - P[E_{S^{(1)}_{f_1}}, E_{S^{(1)}_{f_2}}, E_{S^{(1)}_{f_3}}] = 0.22$ . In this case,

$C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 6.90$  (3.57+3.33). Similarly, for  $f_a^{(1),c} = \{f_1, f_3\}$ ,  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 4.15$  and for

$f_a^{(1),c} = \{f_2, f_3\}$ ,  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 4.15$ . For  $f_a^{(1),c} = \{f_1, f_2, f_3\}$ ,  $f_y^{(1),c} = \{\phi\}$ ,  $DS_{f_a^{(1),c}}^{(1)} = \{D_1, D_2\}$ ,

$DS_{f_y^{(1),c}}^{(1)} = \{\phi\}$ , and  $\beta_{f_a^{(1),c}}^{(1),c} = P[E_{S^{(1)}_{f_1}}, E_{S^{(1)}_{f_2}}, E_{S^{(1)}_{f_3}}] = 0.01$ . In this case,  $C_{f_a^{(1),c}, DS_{f_a^{(1),c}}^{(1)}}^{(1),c} = 4.15$ .

Using (3.1), the multicasting cost for  $f_x^{(1),c} = \{f_1, f_2, f_3\}$  is  $C_{S^{(1)}, DS^{(1)}}^{(1),c} = 8.04$ . The multicasting cost for other subsets of  $\{f_1, f_2, f_3\}$  which reaches  $DS^{(1)}$  can be similarly calculated.

---

**Algorithm 2: Simulation** ( $n, DS_n^{(1)}, FS_n^{(1)}, C_{n, DS_n^{(1)}}^{(1),c}, count$ )

---

1.  $k = 1; s_0 = 0; fs_n^{(1),c} = \{\phi\}; count = 0; DS_{us} = DS_n^{(1)};$   
 $F_n^{(1)}[1] = F_n^{(1)}[2] = \dots = \phi; D_n^{(1)}[1] = D_n^{(1)}[2] = \dots = \phi;$
  2. **for** each  $f_b \subseteq FS_n^{(1)}$ , where  $|f_b| = \{0, \dots, |FS_n^{(1)}|\}$  **do**
  3. compute  $\beta_{f_a^{(1),c}}^{(1),c}$  in (3.1) for  $f_b$ , represent it as  $\beta_{f_a^{(1),c}}^{(1),c}(f_b)$
  4.  $s_k = s_{k-1} + \beta_{f_a^{(1),c}}^{(1),c}(f_b)$ , where  $s_k \in [0, 1]$
  5.  $k = k + 1$ ;
  6. Generate *rand*
  7.  $count = count + 1$
  8. **if**  $s_{k-1} < rand \leq s_k$ , where  $k = 1, \dots, 2^{|FS_n^{(1)}|}$  **then**
  9.  $f_b$  corresponding to  $(s_{k-1}, s_k]$  is noted and  $fs_n^{(1),c} = fs_n^{(1),c} \cup f_b$ ;
  10.  $DS_{us} = DS_{us} - DS_{fs_n^{(1),c}}^{(1)}$
  11. **if**  $DS_{us} \neq \{\phi\}$  **then**
  12. go to 6 ;
  13. **for** each  $f_u \in fs_n^{(1),c}$  **do**
  14. **for** each  $D_{f_u} \subseteq DS_n^{(1)}$  **do**
  15. Find nodes  $f_u$ 's and the corresponding  $D_{f_u}$ 's, such that the overall cost to reach from  $fs_n^{(1),c}$  to  $DS_n^{(1)}$  ( $C_{fs_n^{(1),c}, DS_n^{(1)}}^{(1),c}$ ) is minimum ;;
  16. Store in sequence all  $f_u$ 's and its corresponding  $D_{f_u}$ 's in  $F_n^{(1)}$  and  $D_n^{(1)}$ , respectively
  17. **for**  $j = 1$  to  $|F_n^{(1)}|$  **do**
  18. **if**  $D_n^{(1)}[j] \neq \{\phi\}$  **then**
  19.  $n = F_n^{(1)}[j], DS_n^{(1)} = D_n^{(1)}[j]$ , and **go to 1** ;;
-

For the given network, it is found that  $f_x^{(l),c} = \{f_1, f_2, f_3\}$  is the forwarder set which has the least multicasting cost of 8.04 to  $DS^{(l)}$ . In case of independent links, the least multicasting cost is 9.80 with  $\{f_1, f_2, f_3\}$  as the forwarder set. When the probabilities of correlation between different links are assumed to be  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_2}] = 0.19$ ,  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_3}] = 0.19$ ,  $P[E_{S^{(l)},f_2}, E_{S^{(l)},f_3}] = 0.19$ ,  $P[E_{S^{(l)},f_1}, E_{S^{(l)},f_2}, E_{S^{(l)},f_3}] = 0.18$ ,  $P[E_{f_3,D_1}, E_{f_3,D_2}] = 0.24$ , the least multicasting cost is 9.15 with  $\{f_3\}$  as the forwarder set. The multicasting cost is different when correlations between the links are considered, and the corresponding forwarder set may also change in some cases.

In Algorithm 1, the pseudo-code for route cost calculation is given. Using this, the forwarder set which provides minimum costs from  $S^{(l)}$  to subsets of  $DS^{(l)}$  is obtained. The forwarder set selection requires knowledge of cost to  $DS^{(l)}$  at all the nodes. A Distributed Bellman-Ford type approach is used to calculate cost at all the network nodes. At first, the cost from the immediately upstream nodes of  $DS^{(l)}$  to all subsets of  $DS^{(l)}$  is calculated while considering the link correlation. Later, immediately upstream nodes of  $DS^{(l)}$  will be considered as a forwarder set which assists the upstream nodes directly connected to them to reach  $DS^{(l)}$ . This backtracking of the nodes will continue until the least cost from all the nodes in the network to the subsets of  $DS^{(l)}$  be known. For the least cost calculation in a network, the link cost and the probability of the correlation are assumed to be known. Any changes in the multicast routes or their costs would initiate a recalculation of the least cost at all the affected nodes, which would compensate for dynamic changes in the network. Routing loops in a network can be avoided by using a split-horizon type approach [5]. However, these loops mostly get eliminated due to their higher cost. Using the multicast least cost algorithm,  $FS^{(l)}$  and least cost from all the nodes of the network to  $DS^{(l)}$  and subsets of  $DS^{(l)}$  is

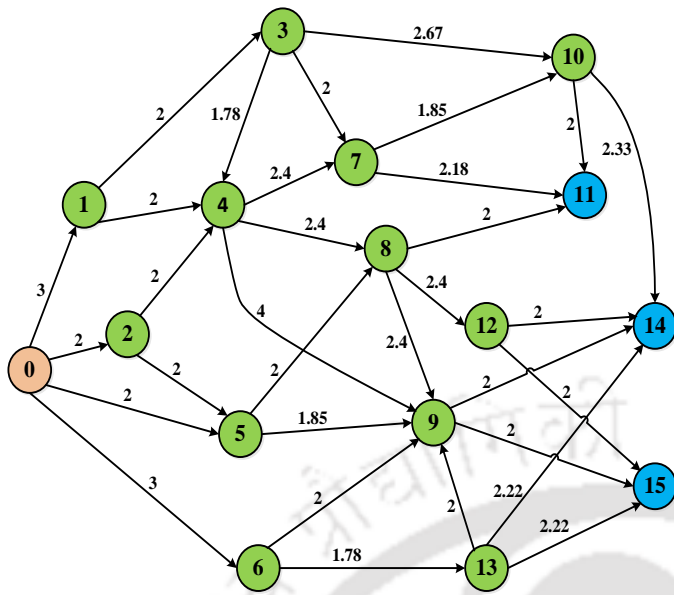
obtained. This assists in packet transmission through a least cost path of the network and provides us the number of transmissions required for successful reception of the packet at the nodes of  $DS^{(l)}$ .

### 3.4 Packet Transmission Strategy

In this section, a packet transmission scheme for multicasting over a network with correlated links is presented. The least cost path information is first obtained at the source  $S^{(l)}$  using the analysis presented in Section 3.3. A packet transmitted by  $S^{(l)}$  may be received by the nodes within the coverage range of  $S^{(l)}$ . The packet received is processed only by the forwarder nodes  $FS^{(l)}$  and is discarded by the other nodes. The source node broadcasts a packet until forwarder nodes, which can reach all destinations, receive the packet successfully. If, during this process, the packet is successfully received by some of the forwarding nodes, which can reach subset of  $DS^{(l)}$ , then they discard further reception, and the packet is targeted for forwarding nodes which cover the remaining destinations. Nodes in  $FS^{(l)}$  acknowledge  $S^{(l)}$  after the successful reception of the packet. If a node belongs to both  $FS^{(l)}$  and  $DS^{(l)}$ , then the node keeps the packet, and forwards a copy of the packet to the intended nodes of  $DS^{(l)}$ . Packet transmission from  $FS^{(l)}$  to  $DS^{(l)}$  is done by searching over all the possible combinations of the nodes in  $FS^{(l)}$  which can reach  $DS^{(l)}$  with the minimum possible cost. The Pseudo-code to simulate packet transmission is presented in Algorithm 2.

### 3.5 Multicasting with Correlated Links: Example

A network is considered where link costs and correlations between them are randomly assumed, as shown in Fig. 3.4. The different source nodes are represented by  $S^{(l)}$  and  $DS^{(l)}$  correspond to different multicast destination sets out of nodes 11, 14, and 15. The cost of transmitting a packet from the source node  $S^{(l)}$  to the destination set  $DS^{(l)}$  is computed using



- $P[E_{0,1}, E_{0,2}] = 0.30$
- $P[E_{0,1}, E_{0,6}] = 0.0333$
- $P[E_{0,2}, E_{0,6}] = 0.0333$
- $P[E_{1,3}, E_{1,4}] = 0.4667$
- $P[E_{3,4}, E_{3,7}] = 0.4375$
- $P[E_{3,7}, E_{3,10}] = 0.0625$
- $P[E_{4,7}, E_{4,9}] = 0.0417$
- $P[E_{5,8}, E_{5,9}] = 0.0417$
- $P[E_{7,10}, E_{7,11}] = 0.0417$
- $P[E_{8,9}, E_{8,12}] = 0.375$
- $P[E_{9,14}, E_{9,15}] = 0.0417$
- $P[E_{12,14}, E_{12,15}] = 0.0417$
- $P[E_{13,9}, E_{13,15}] = 0.10$
- $P[E_{0,1}, E_{0,2}, E_{0,5}] = 0.2667$
- $P[E_{0,1}, E_{0,5}, E_{0,6}] = 0.0333$
- $P[E_{3,4}, E_{3,7}, E_{3,10}] = 0.0625$
- $P[E_{8,9}, E_{8,11}, E_{8,12}] = 0.0417$
- $P[E_{0,1}, E_{0,2}, E_{0,5}, E_{0,6}] = 0.0333$
- $P[E_{0,1}, E_{0,5}] = 0.30$
- $P[E_{0,2}, E_{0,5}] = 0.4667$
- $P[E_{0,5}, E_{0,6}] = 0.0333$
- $P[E_{2,4}, E_{2,5}] = 0.4667$
- $P[E_{3,4}, E_{3,10}] = 0.0625$
- $P[E_{4,7}, E_{4,8}] = 0.0417$
- $P[E_{4,8}, E_{4,9}] = 0.21$
- $P[E_{6,9}, E_{6,13}] = 0.0625$
- $P[E_{8,9}, E_{8,11}] = 0.0417$
- $P[E_{8,11}, E_{8,12}] = 0.0417$
- $P[E_{10,11}, E_{10,14}] = 0.0714$
- $P[E_{13,9}, E_{13,14}] = 0.10$
- $P[E_{13,14}, E_{13,15}] = 0.40$
- $P[E_{0,1}, E_{0,2}, E_{0,6}] = 0.0333$
- $P[E_{0,2}, E_{0,5}, E_{0,6}] = 0.0333$
- $P[E_{4,7}, E_{4,8}, E_{4,9}] = 0.0417$
- $P[E_{13,9}, E_{13,14}, E_{13,15}] = 0.10$

Figure 3.4: Example network 3: Multicasting

Table 3.1: Comparison of multicast route cost and selected forwarder set for an example network 3 having independent links and correlated links.

Source (S <sup>(d)</sup> )	Destination Set (DS <sup>(d)</sup> )							
	{11, 14}		{11, 15}		{14, 15}		{11, 14, 15}	
	Independent links	Correlated links	Independent links	Correlated links	Independent links	Correlated links	Independent links	Correlated links
0	7.62 (1,2,5)	8.41 ( <b>8.41</b> ) (1,5)	7.89 (2,5)	8.49 ( <b>8.48</b> ) (2,5)	5.72 (5,6)	5.58 ( <b>5.58</b> ) (5,6)	8.56 (2,5)	9.45 ( <b>9.44</b> ) (2,5)
1	6.62 (3,4)	7.10 ( <b>7.10</b> ) (3,4)	8.25 (3,4)	8.78 ( <b>8.87</b> ) (3,4)	7.21 (3,4)	8.12 ( <b>8.13</b> ) (3,4)	8.92 (3,4)	9.74 ( <b>9.83</b> ) (3,4)
2	7.20 (4,5)	7.73 ( <b>7.73</b> ) (4,5)	7.50 (4,5)	8.40 ( <b>8.40</b> ) (4,5)	6.04 (4,5)	6.77 ( <b>6.77</b> ) (4,5)	8.17 (4,5)	9.35 ( <b>9.36</b> ) (4,5)
3	5.06 (7,10)	5.18 ( <b>5.18</b> ) (7,10)	8.10 (4)	8.58 ( <b>8.66</b> ) (4)	7.06 (4)	7.91 ( <b>7.91</b> ) (4)	8.77 (4)	9.53 ( <b>9.62</b> ) (4)
4	5.76 (7,8,9)	5.81 ( <b>5.81</b> ) (7,8,9)	6.33 (7,8,9)	6.80 ( <b>6.88</b> ) (8,9)	5.28 (8,9)	6.13 ( <b>6.13</b> ) (8,9)	6.99 (7,8,9)	7.75 ( <b>7.84</b> ) (8,9)
5	6.09 (8,9)	6.50 ( <b>6.50</b> ) (8,9)	6.09 (8,9)	6.50 ( <b>6.51</b> ) (8,9)	4.41 (8,9)	4.81 ( <b>4.80</b> ) (9)	6.75 (8,9)	7.46 ( <b>7.46</b> ) (8,9)
6	∞	∞	∞	∞	4.01 (9,13)	3.66 ( <b>3.66</b> ) (9,13)	∞	∞
7	4.41 (10,11)	4.61 ( <b>4.61</b> ) (10,11)	∞	∞	∞	∞	∞	∞
8	4.31 (9,11,12)	5.10 ( <b>5.10</b> ) (9,10,11)	4.31 (9,11,12)	5.10 ( <b>5.10</b> ) (9,11,12)	4.18 (9,12)	5.14 ( <b>5.14</b> ) (9,12)	4.97 (9,11,12)	6.05 ( <b>6.05</b> ) (9,11,12)
9	∞	∞	∞	∞	2.67 (14,15)	2.96 ( <b>2.95</b> ) (14,15)	∞	∞
10	2.93 (11,14)	3.16 ( <b>3.16</b> ) (11,14)	∞	∞	∞	∞	∞	∞
12	∞	∞	∞	∞	2.67 (14,15)	2.96 ( <b>2.95</b> ) (14,15)	∞	∞
13	∞	∞	∞	∞	2.87 (9,14,15)	2.42 ( <b>2.43</b> ) (14,15)	∞	∞

algorithm 1. The results are tabulated in Table 3.1 and are compared with the network where the links are assumed to be independent. It is observed from Table 3.1 that the costs from  $S^{(l)}$  to the destination set  $DS^{(l)}$  for the two cases are different. To confirm our results, the packet transmission is performed using algorithm 2. Each simulation is run for 1 million packets, and the average number of transmissions needed to deliver a packet from source to destination set reliably is calculated. The six independent runs are done to evaluate confidence levels and confidence intervals to confirm that the experimental results are statistically significant. The results are within 5% of the mean value with a confidence level of at least 95%. Simulation results for multicasting considering link correlation are mentioned next to analytical results in bold for comparison.

### 3.6 Summary

In this chapter, an analytical expression to compute cost in case of multicasting with correlated links is presented. Costs calculated using M-LCAR [5] with link correlations taken into account are employed as input to decide the packet transmission strategy. It is observed that multicasting costs to reach destination subsets are significantly different on considering the link correlations when compared to the network with independent links where the *ETX* costs of the network links are considered to be the same. It is also observed that the least cost route to the destination subset also changes in some cases. Analytical results obtained for the multicast network with correlated links are verified through simulations.





## Chapter 4

# Network Coding Assisted Reliable Multicasting in Multi-Hop Wireless Networks

In the previous chapter, the impact of link correlations on multicast route cost calculation is shown and a packet transmission strategy proposed for networks with correlated links. This chapter presents a NC-MLCAR scheme for packet transmission over a multi-hop wireless network for both independent links and for a correlated links environment. The extensive simulation results show that the proposed NC-MLCAR scheme significantly reduces the expected number of overall transmissions in the network for independent and correlated links compared to the existing multicasting schemes.

### 4.1 Introduction

NC takes into account the information content of each transmission and uses that knowledge to form a network-coded packet. In this chapter, a binary NC scheme for packet transmission is considered, which has simpler coding and decoding structure when compared to the complex RLNC scheme [4]. The objective is to transmit packets efficiently from the source node to the destination set with the help of a selected forwarder set to reduce the ETXs for efficient network capacity usage. The forwarder set is selected using MLCAR algorithm [5]. Network-coded packets are then transmitted by allocating traffic efficiently between the nodes of the forwarder set, while ensuring that each destination receives all the packets. The complexity analysis of NC-MLCAR is presented and compared with the corresponding MLCAR scheme.

The rest of the chapter is organized as follows. In Section 4.2, the implementation details of NC-MLCAR are presented. Section 4.3 presents the proposed algorithm to transmit packets from the source to the destination set efficiently. In Section 4.4, simulation results and complexity analysis of NC-MLCAR is introduced. Finally, Section 4.5 concludes the chapter.

## 4.2 Implementation Details: NC-MLCAR

A stationary multi-hop wireless mesh network is considered where the source node  $S^{(l)}$  multicasts a data file to  $M^{(l)}$  ( $M^{(l)} > 1$ ) destinations  $D_1, \dots, D_{M^{(l)}}$  with the help of  $FS^{(l)}$  selected using the MLCAR algorithm [5]. The data file from  $S^{(l)}$  is transmitted as batches of packets, where each packet is communicated in a time slot of fixed duration. At  $S^{(l)}$ , the data file is divided into batches of  $N^{(l)}$  packets. Let  $B^{(l)}$  be a batch of  $N^{(l)}$  packets, represented by  $B^{(l)} = (P_1^{(l)}, \dots, P_{N^{(l)}}^{(l)})$ . Information regarding successful or unsuccessful reception of a packet is communicated to the sender using ACKs and NACKs, respectively. ACKs/NACKs are assumed to be instantaneous and error-free. Once a batch is successfully received at the first-hop neighbors of the source towards the destined destination nodes, the next batch is forwarded by the source. The network-coded packets are transmitted over a wireless medium. At  $S^{(l)}$ , the coded combinations of packets in a batch  $B^{(l)}$  are formed. The  $k^{\text{th}}$  coded packet is

$P_k^{(l)} = \sum_{i=1}^{N^{(l)}} c_{ki} P_i^{(l)}$ , where  $c_{ki} \in \{0, 1\}$  and  $P_i^{(l)}$  is  $i^{\text{th}}$  packet in  $B^{(l)}$ . The code vector for the  $k^{\text{th}}$

coded packet is  $c_k = (c_{k1}, \dots, c_{kN^{(l)}})$ . Therefore, a network-coded packet  $P_k^{(l)}$  is the XOR combination of packets for which the elements of  $c_k$  are 1. The code vector is decided such that the coded packet is innovative for a maximum number of selected receiving nodes. A packet is said to be an innovative packet for a node if it is linearly independent from the packets, which are already present there. Once all the destinations receive  $N^{(l)}$  innovative

packets (i.e.  $N^{(l)}$  packets with linearly independent binary code vectors), they can retrieve the batch. A coded packet formed by combination of  $J$  packets can be decoded at a node, only if the node has  $J - 1$  innovative packets out of  $J$  packets.

In the proposed scheme, a  $FS^{(l)}$  is selected from the neighboring nodes, such that the cost to  $DS^{(l)}$  is minimum. The multicast cost metric is determined using the *ETX* link cost. Network link cost metric *ETX* is defined as  $1/p_s$ , where  $p_s$  is the probability that packets are successfully transmitted over the link. The probability that the packet is not received at any of the nodes in  $f_x^{(l),i}$  is given by  $P_F$ . The cost to multicast a packet from  $S^{(l)}$  to  $DS^{(l)}$  over an independent link network is given by (4.1)

$$C_{S^{(l)}, DS^{(l)}}^{(l),i}(f_x^{(l),i}) = \frac{P_F + \sum_{\forall f_a^{(l),i} \subseteq f_x^{(l),i}} \beta_{f_a^{(l),i}}^{(l),i} \left( 1 + C_{f_a^{(l),i}, DS^{(l)}}^{(l),i} + C_{S^{(l)}, f_y^{(l),i}}^{(l),i} \right)}{1 - P_F} \quad (4.1)$$

The cost to multicast a packet from  $S^{(l)}$  to  $DS^{(l)}$  over a correlated link network is given by (4.2)

$$C_{S^{(l)}, DS^{(l)}}^{(l),c}(f_x^{(l),c}) = \frac{1 + \sum_{\forall f_a^{(l),c} \subseteq f_x^{(l),c}, f_a^{(l),c} \neq \phi} \beta_{f_a^{(l),c}}^{(l),c} \left( C_{f_a^{(l),c}, DS^{(l)}}^{(l),c} + C_{S^{(l)}, f_y^{(l),c}}^{(l),c} \right)}{\sum_{k=1}^{|f_x^{(l),c}|} (-1)^{k-1} P(f^k)} \quad (4.2)$$

The algorithm for  $FS^{(l)}$  selection in an independent link network is presented in [5]. Chapter 3 introduced the  $FS^{(l)}$  selection algorithm for a network with correlated wireless links. In the NC-MLCAR scheme,  $FS^{(l)}$  selection for the network with both independent and correlated links is done by applying the multicasting cost defined in (4.1) and (4.2), respectively. The index  $i$  and  $c$  are dropped in the subsequent sections for the convenience of notation. The MAC layer multicast protocol discussed in [88] is followed. In the protocol, the

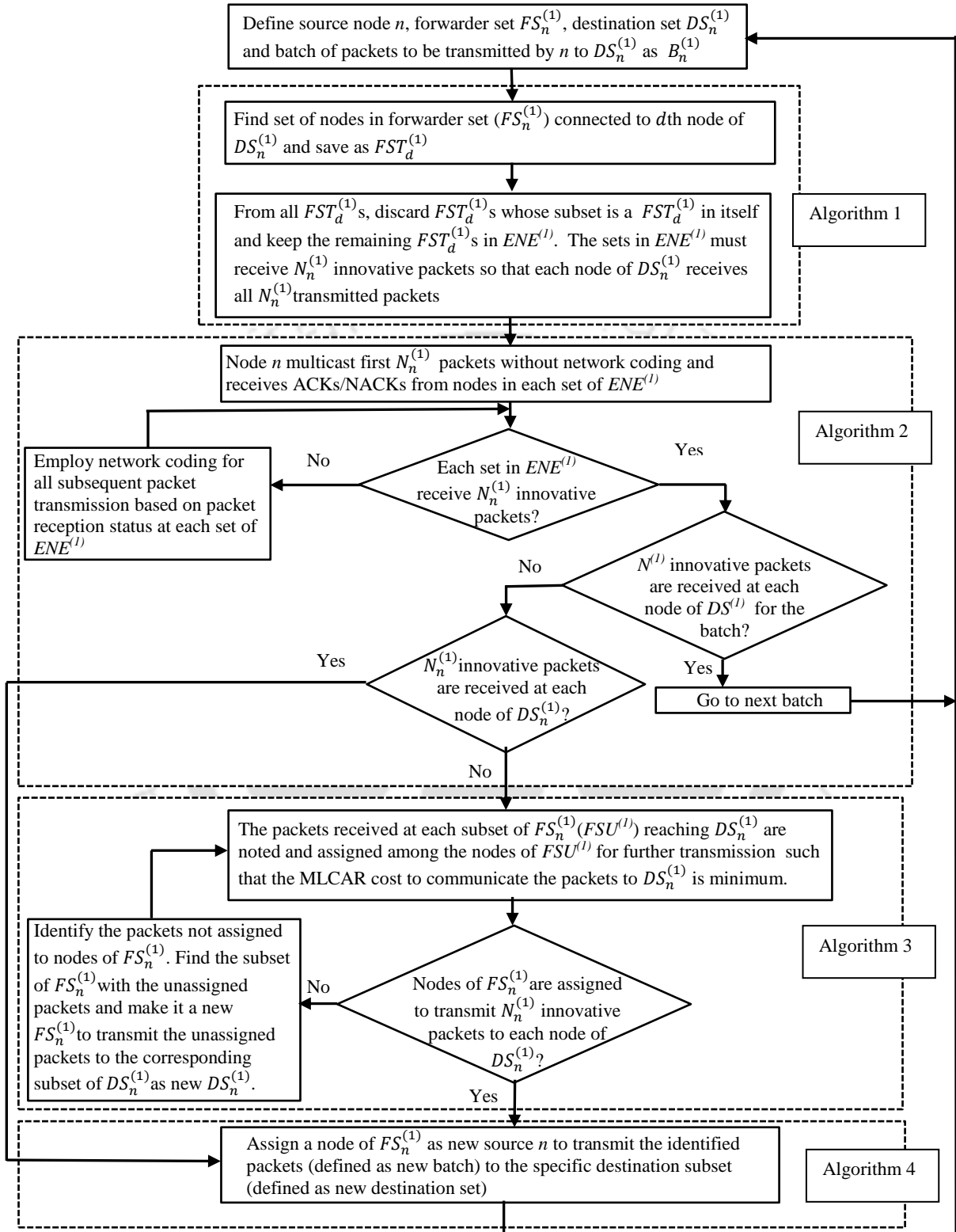


Figure 4.1: Flowchart of NC-MLCAR

signaling overhead to execute reliable multicasting includes *RTS* (Ready to Send), *CTS* (Clear to Send), and *ACK/NACK*. Overhead packets required to transmit the  $i^{\text{th}}$  by the network node  $n$  are  $n_i = RTS + V_n^{(l)}(CTS + ACK/NACK)$ , where  $V_n^{(l)}$  is size of  $FS_n^{(l)}$ .

### 4.3 Proposed Algorithm: NC-MLCAR

In this section, a scheme is presented for efficient transmission of innovative packets from  $n$  to  $DS_n^{(l)}$  with a selected  $FS_n^{(l)}$ . The scheme is presented with the help of four algorithms. In Algorithm 1, subsets of forwarder set which must receive all innovative packets are identified. This is done to ensure that all the destinations can eventually receive  $N_n^{(l)}$  innovative packets. Next, in Algorithm 2,  $N_n^{(l)}$  innovative packets are reliably transmitted to the subsets of the  $FS_n^{(l)}$  obtained in Algorithm 1. This includes deciding the code vector for

---

**Algorithm 1:** Finding subsets of  $FS_n^{(l)}$  which must have  $N_n^{(l)}$  innovative packets

---

**Initial Conditions:**  $n$ ;  $FS_n^{(l)} = \{f_1, \dots, f_{V_n^{(l)}}\}$ ;  $DS_n^{(l)} = \{D_1, \dots, D_{M_n^{(l)}}\}$ ;

$FST_d^{(l)} = \phi$  for  $d \in DS_n^{(l)}$ ;  $AFST^{(l)}[1] = \dots = AFST^{(l)}[|AFST^{(l)}|] = \phi$ ;

**Output:**  $ENE^{(l)}$ ;  $K^{(l)} = \text{size}(ENE^{(l)})$ ;

1. **for** each  $d \in DS_n^{(l)}$  **do**
  2.     **for** each  $f \in FS_n^{(l)}$  **do**
  3.         **if**  $C_{f,d}^{(l),w} < \infty$ ,  $w \in (i, c)$  **then**
  4.              $FST_d^{(l)} = FST_d^{(l)} \cup f$
  5.         **end if**
  6.     **end for**
  7. **end for**
  8.  $AFST^{(l)} =$  set of  $FST_d^{(l)}$ 's sorted in ascending order of their size
  9. **for**  $i = 1$  to  $M_n^{(l)}$  **do**
  10.     **for**  $j = (i + 1)$  to  $M_n^{(l)}$  **do**
  11.         **if**  $AFST^{(l)}[i] \neq \phi$ ,  $AFST^{(l)}[j] \neq \phi$ ,  $AFST^{(l)}[i] \subseteq AFST^{(l)}[j]$  **then**
  12.              $AFST^{(l)}[j] = \phi$
  13.         **end if**
  14.     **end for**
  15. **end for**
  16.  $ENE^{(l)} = AFST^{(l)}$  excluding all null elements
-

more efficient transmission. Algorithm 3 does packet distribution at the nodes of the  $FS_n^{(l)}$ , so that each destination receives  $N_n^{(l)}$  innovative packets. After traffic distribution in Algorithm 3,  $n$ ,  $DS_n^{(l)}$ , and  $FS_n^{(l)}$  are updated in Algorithm 4 to forward the packets to the destination nodes. Nodes of the  $FS_n^{(l)}$  in Algorithm 4 act as the new source to transmit the assigned packets further to their corresponding destination set. The proposed algorithm is shown with the help of a flowchart in Fig. 4.1.

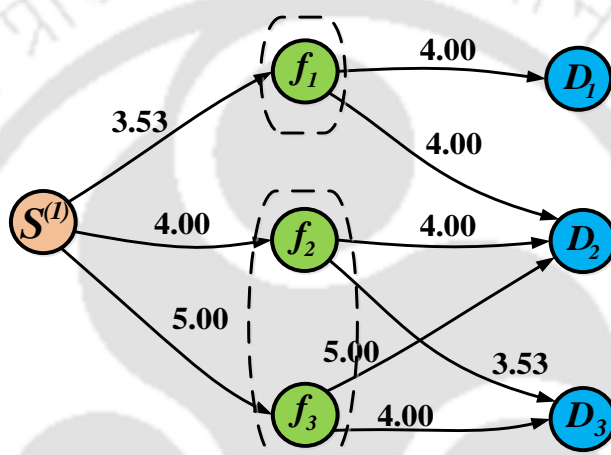


Figure 4.2: Example network 1: Multicasting

To follow the steps of the proposed algorithms, a simple multicast network with independent links shown in Fig. 4.2 is considered. Here  $S^{(l)}$  multicasts a batch of 5 packets  $(P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  to  $DS^{(l)} = \{D_1, D_2, D_3\}$ . The  $ETX$  cost of a link are mentioned against it in the figure. Using MLCAR algorithm,  $FS^{(l)} = \{f_1, f_2, f_3\}$  is identified as the  $FS^{(l)}$  with the least cost to  $DS^{(l)}$ . The selected  $FS^{(l)}$  is used to communicate packets from node  $S^{(l)}$  to  $DS^{(l)}$  using Algorithms 1 - 4. To start with, the variables  $n$ ,  $FS_n^{(l)}$ ,  $DS_n^{(l)}$ ,  $N_n^{(l)}$ ,  $V_n^{(l)}$ ,  $M_n^{(l)}$ , and  $IP_n^{(l)}$  are defined as  $n = S^{(l)}$ ,  $FS_n^{(l)} = FS^{(l)}$ ,  $DS_n^{(l)} = DS^{(l)}$ ,  $N_n^{(l)} = N^{(l)}$ ,  $V_n^{(l)} = V^{(l)}$ ,  $M_n^{(l)} = M^{(l)}$  and  $IP_n^{(l)} = B^{(l)}$ .

In Algorithm 1 (lines 1 - 7), nodes in  $FS_n^{(l)}$  connected to  $p^{\text{th}}$  node of  $DS_n^{(l)}$  are identified and stored in  $FST_d^{(l)}$  for  $d = \{1, \dots, M_n^{(l)}\}$ . For the network in Fig. 4.2,  $FST_{D_1}^{(l)} = \{f_1\}$ ,  $FST_{D_2}^{(l)} = \{f_1, f_2, f_3\}$ , and  $FST_{D_3}^{(l)} = \{f_2, f_3\}$ . Next,  $FST_d^{(l)}$ 's are sorted in ascending order of their size and stored in  $AFST^{(l)}$  (line 8). This gives  $AFST^{(l)} = \{\{f_1\}, \{f_2, f_3\}, \{f_1, f_2, f_3\}\}$ . Since each element of  $AFST^{(l)}$  communicates packets to a specific node of  $DS_n^{(l)}$ ; each element of  $AFST^{(l)}$  must receive all  $N_n^{(l)} (= 5)$

---



---

**Algorithm 2:** Packet Transmission

---

**Initial Conditions:**  $IP_n^{(l)} = (P_1^{(l)}, \dots, P_{N_n^{(l)}}^{(l)})$ ;

**Input:**  $ENE^{(l)}$ ;  $K^{(l)}$ ;  $N_n^{(l)}$ ;  $FS_n^{(l)}$ ;

**Output:** Packet reception status at nodes of  $FS_n^{(l)}$ ;

1. Node  $n$  multicast packets  $IP_n^{(l)}$  as first  $N_n^{(l)}$  transmissions
  2. Nodes in  $FS_n^{(l)}$  sends packet reception status with the help of ACKs/NACKs to  $n$
  3. **if**  $K^{(l)} = 1$  **then**
  4.  $n$  transmits missing packets without network coding until nodes of  $ENE^{(l)}$  collectively receives  $N_n^{(l)}$  innovative packets
  5. **end if**
  6. **if**  $K^{(l)} > 1$  and packet is missing at all the nodes of  $ENE^{(l)}[k]$ ,  $k = 1, \dots, K^{(l)}$  **then**
  7.  $n$  transmits missing packets without network coding until the packet is successfully received by at least one node of any  $ENE^{(l)}[k]$ ,  $k = 1, \dots, K^{(l)}$
  8. **end if**
  9. **if**  $K^{(l)} > 1$  and packet is missing at the nodes of  $ENE^{(l)}[k]$ ,  $k = 1, \dots, K^{(l)}$  **then**
  10.  $n$  multicast XOR combinations of disjoint packets missing at more than one  $ENE^{(l)}[k]$ s. If packets are missing at only one  $ENE^{(l)}[k]$  or common packets are missing at  $ENE^{(l)}[k]$ s,  $n$  transmits packets without network coding. The transmission continues until nodes of  $ENE^{(l)}[k]$ ,  $k = 1, \dots, K^{(l)}$ , collectively receives  $N_n^{(l)}$  innovative packets
  11. **end if**
  12. **if**  $N_n^{(l)}$  innovative packets are received at all nodes  $D_1, \dots, D_{M_n^{(l)}}$  **then**
  13. Initialize:  $n = S^{(l)}$ ,  $FS_n^{(l)} = FS^{(l)}$ ,  $DS_n^{(l)} = DS^{(l)}$ ,  $N_n^{(l)} = N^{(l)}$ ,  $V_n^{(l)} = V^{(l)}$ ,  $M_n^{(l)} = M^{(l)}$ , and  $IP_n^{(l)} = B^{(l)}$ , and switch to next batch and **go to** step 1 of Algorithm 1
  14. **else if**  $N_n^{(l)}$  innovative packets are received at all nodes  $D_1, \dots, D_{M_n^{(l)}}$  **then**
  15. **go to** step 1 of Algorithm 4
  16. **end if**
- 
-

innovative packets. An element of  $AFST^{(l)}$  may be a subset of another element of  $AFST^{(l)}$ , and since both the elements receive all the  $N_n^{(l)}$  innovative packets, the element with a bigger set size is discarded. The remaining elements are stored in  $ENE^{(l)}$  (lines 9 - 16). For the network in Fig. 4.2,  $ENE^{(l)} = \{\{f_1\}, \{f_2, f_3\}\}$  and  $K^{(l)} = 2$ .

Next, Algorithm 2 is followed for packet transmission from  $n$  to  $FS_n^{(l)}$  until each element of  $ENE^{(l)}$  receives  $N_n^{(l)}$  innovative packets ( $IP_n^{(l)}$ ). At first (line 1), node  $n$  multicast  $N_n^{(l)}$  packets of the batch without NC. Once node  $n$  receives the status of the packets it has transmitted, it can decide the code vector for its subsequent transmissions. After first  $N_n^{(l)}$  transmissions, a coded combination of packets is sent from the source only if the coded packet is innovative for more than one element of  $ENE^{(l)}$ ; otherwise an uncoded (native) packet is transmitted (lines 3 - 11). For the network example of Fig. 4.2, suppose that the packets received after the first 5 transmissions at  $FS_n^{(l)}$  nodes  $f_1, f_2$ , and  $f_3$  are  $(P_1^{(l)}, P_3^{(l)}, P_4^{(l)})$ ,  $(P_2^{(l)}, P_5^{(l)})$ , and  $(P_1^{(l)}, P_5^{(l)})$ , respectively. Then, packets received at  $\{f_1\}$  and  $\{f_2, f_3\}$  are  $(P_1^{(l)}, P_3^{(l)}, P_4^{(l)})$  and  $(P_1^{(l)}, P_2^{(l)}, P_5^{(l)})$ , respectively. Since each element of  $ENE^{(l)}$  has not received 5 innovative packets,  $n$  transmit combination of packets which are innovative for more than one element of  $ENE^{(l)}$ . Suppose,  $n$  transmits  $(P_2^{(l)} \oplus P_3^{(l)})$  and the packet is successfully received at  $f_1$  and  $f_2$  but not at  $f_3$ . The updated packet reception status for forwarders  $f_1, f_2$ , and  $f_3$  are then  $(P_1^{(l)}, P_3^{(l)}, P_4^{(l)}, P_2^{(l)} \oplus P_3^{(l)})$ ,  $(P_2^{(l)}, P_5^{(l)}, P_2^{(l)} \oplus P_3^{(l)})$ , and  $(P_1^{(l)}, P_5^{(l)})$  respectively.  $f_1$  can extract  $(P_2^{(l)})$  using  $(P_3^{(l)})$  and  $(P_2^{(l)} \oplus P_3^{(l)})$  and  $f_2$  can extract  $(P_3^{(l)})$  using  $(P_2^{(l)})$  and  $(P_2^{(l)} \oplus P_3^{(l)})$ . Again a coded packet  $(P_4^{(l)} \oplus P_5^{(l)})$  is multicast from  $n$ . On successful reception, the packet status at  $f_1, f_2$ , and  $f_3$  is updated to  $f_1 : (P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ ,  $f_2 : (P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ ,

$f_3 : (P_1^{(1)}, P_4^{(1)}, P_5^{(1)})$ . Thus, intermediate nodes of  $ENE^{(1)}[1]$  and  $ENE^{(1)}[2]$  have received 5 innovative packets. In order to further communicate the packets to

---

**Algorithm 3:** Traffic Distribution

---

**Initial Conditions:**  $F_n^{(1)}[1] = F_n^{(1)}[2] = \dots = \phi$ ;  $D_n^{(1)}[1] = D_n^{(1)}[2] = \dots = \phi$ ;  
 $IP_n^{(1)}[1] = IP_n^{(1)}[2] = \dots = \phi$ ;

**Input Conditions:** Packet reception status at nodes of  $FS_n^{(1)}$ ;  $N_n^{(1)}$ ;  $FS_n^{(1)}$ ;  $DS_n^{(1)}$ ;

**Output:**  $F_n^{(1)}$ ;  $D_n^{(1)}$ ;  $IP_n^{(1)}$ ;

1. **for**  $l = 1$  to  $|FSU^{(1)}|$  **do**
  2.   **for** each  $f_p \in FSU^{(1)}[l]$  **do**
  3.     **for** each  $DSU_{f_p}^{(1)}$  **do**
  4.       Find nodes  $f_p$ 's and the corresponding  $DSU_{f_p}^{(1)}$ 's, such that the overall cost to reach from  $FSU^{(1)}[l]$  to  $DS_n^{(1)}$  ( $C_{FSU^{(1)}[l], DS_n^{(1)}}^{(1),w}$ ,  $w \in \{i, c\}$ ) is minimum
  5.       **end for**
  6.     **end for**
  7.     Store in sequence all  $f_p$ 's and  $DSU_{f_p}^{(1)}$ 's in  $F_n^{(1)}$  and  $D_n^{(1)}$ , respectively
  8.     Innovative packets successfully received at all the nodes of  $FSU^{(1)}[l]$  and not at nodes of  $FS_n^{(1)} - FSU^{(1)}[l]$  (packets may have different code vector) are stored against the corresponding  $F_n^{(1)}$  and  $D_n^{(1)}$  in  $IP_n^{(1)}$ . Delete the null elements of  $IP_n^{(1)}$  and its corresponding  $F_n^{(1)}$  and  $D_n^{(1)}$ , if any
  9.     **end for**
  10.   **if** nodes of  $FS_n^{(1)}$  are assigned to communicate  $N_n^{(1)}$  innovative packets to each node of  $DS_n^{(1)}$  **then**
  11.     **continue**
  12.   **else**
  13.     Find the unassigned packets at the nodes of  $FS_n^{(1)}$  and the corresponding nodes of  $DS_n^{(1)}$  with the unassigned packets. Find the  $FS_n^{(1)}$  subset with these unassigned packets. Make this  $FS_n^{(1)}$  subset a new  $FS_n^{(1)}$  to transmit unassigned packets to the identified subset of  $DS_n^{(1)}$  as new  $DS_n^{(1)}$  and **go to** step 1 of Algorithm 3
  14.   **end if**
  15.   Sort elements of  $D_n^{(1)}$  in ascending order of size and arrange  $F_n^{(1)}$  and  $IP_n^{(1)}$  based on modified index order of  $D_n^{(1)}$  on sorting
- 

$D_1$ ,  $D_2$ , and  $D_3$  (lines 12 - 16), Algorithm 3 is followed.

---

**Algorithm 4: Initialization**

---

**Input Conditions:**  $F_n^{(1)}$ ;  $D_n^{(1)}$ ;  $IP_n^{(1)}$ ;  $FS_n^{(1)}$ ;**Output:**  $n$ ;  $DS_n^{(1)}$ ;  $IP_n^{(1)}$ ;  $N_n^{(1)}$ ;

1. **for** each  $f_u \in FS_n^{(1)}$  **do**
  2.  $i = 1; j = 1; DE_{f_u}^{(1)}[1] = DE_{f_u}^{(1)}[2] = \dots = \phi; PE_{f_u}^{(1)}[1] = PE_{f_u}^{(1)}[2] = \dots = \phi;$
  3. **for** each  $f_v \in F_n^{(1)}$  **do**
  4. **if**  $f_u == f_v$  **then**
  5. **if**  $DE_{f_u}^{(1)}[1] == \phi$  **then**
  6.  $DE_{f_u}^{(1)}[1] = DE_{f_u}^{(1)}[1] \cup D_n^{(1)}[v]; PE_{f_u}^{(1)}[1] = PE_{f_u}^{(1)}[1] \cup IP_n^{(1)}[v]$
  7. **end if**
  8. **if**  $j > 1$  **then**
  9.  $DE_{f_u}^{(1)}[j] = DE_{f_u}^{(1)}[j] \cup D_n^{(1)}[v]$
  10. **for**  $i = 1$  to  $j-1$  **do**
  11. **if**  $DE_{f_u}^{(1)}[j] \cap DE_{f_u}^{(1)}[i] = DE_{f_u}^{(1)}[i], DE_{f_u}^{(1)}[i] \neq \phi$  **then**
  12.  $PE_{f_u}^{(1)}[i] = PE_{f_u}^{(1)}[i] \cup IP_n^{(1)}[v]$
  13. **end if**
  14. **if**  $(DE_{f_u}^{(1)}[j] \cup DE_{f_u}^{(1)}[i]) - DE_{f_u}^{(1)}[i] \neq \phi, DE_{f_u}^{(1)}[i] \neq \phi$  **then**
  15.  $DE_{f_u}^{(1)}[j] = (DE_{f_u}^{(1)}[j] \cup DE_{f_u}^{(1)}[i]) - DE_{f_u}^{(1)}[i]$
  16.  $PE_{f_u}^{(1)}[j] = PE_{f_u}^{(1)}[j] \cup IP_n^{(1)}[v]$
  17. **else**
  18.  $DE_{f_u}^{(1)}[j] = \phi, PE_{f_u}^{(1)}[j] = \phi$
  19. **end if**
  20. **end for**
  21. **end if**
  22.  $j = j + 1$
  23. **end if**
  24. **end for**
  25. Remove null elements of  $DE_{f_u}^{(1)}$  and  $PE_{f_u}^{(1)}$ . Set  $t^{(1)} = |DE_{f_u}^{(1)}| = |PE_{f_u}^{(1)}|$ .
  26. **if**  $t^{(1)} > 0$  **then**
  27. Initialize:  $n = f_u, DS_n^{(1)} = \{DE_{f_u}^{(1)}[1], \dots, DE_{f_u}^{(1)}[t^{(1)}]\},$   
 $IP_n^{(1)} = \{IP_n^{(1)} \cup PE_{f_u}^{(1)}[1], \dots, IP_n^{(1)} \cup PE_{f_u}^{(1)}[t^{(1)}]\},$   
 $N_n^{(1)} = \max\{\text{size}(PE_{f_u}^{(1)}[1]), \dots, \text{size}(PE_{f_u}^{(1)}[t^{(1)}])\}$  and **go to** step 1 of Algorithm 1
  28. **else**
  29. **go to** step 1 of Algorithm 4
  30. **end if**
  31. **end for**
-

In Algorithm 3, the multicasting cost  $C_{f_p, DSU_{f_p}}^{(l),w}$  of independent and correlated link network computed using (4.1) and (4.2), respectively is fetched for each node of  $FSU^{(l)}[l]$ . The  $f_p$ 's ( $f_p \in FSU^{(l)}[l]$ ) and corresponding  $DSU_{f_p}^{(l)}$ 's are identified, such that the multicasting cost from  $FSU^{(l)}[l]$  to reach  $DS_n^{(l)}$  is minimum. The nodes  $f_c$ 's of  $FSU^{(l)}[l]$  and the corresponding  $DSU_{f_p}^{(l)}$ 's are stored in  $F_n^{(l)}$  and  $D_n^{(l)}$ , respectively. Next, the packet reception status at the nodes of  $FS_n^{(l)}$  obtained in Algorithm 2 is used to find the packets received at all nodes of  $FSU^{(l)}[l]$  and missing at nodes of  $FS_n^{(l)} - FSU^{(l)}[l]$ . The packets thus found for each  $FSU^{(l)}[l]$  are stored in  $IP_n^{(l)}$  for the corresponding elements of  $F_n^{(l)}$  and  $D_n^{(l)}$  (lines 1 - 9). Consider the network in Fig. 4.2 with  $FSU^{(l)} = \{\{f_1, f_2\}, \{f_1, f_3\}, \{f_1, f_2, f_3\}\}$ . The cost from  $FSU^{(l)}[1]$  to  $DS_n^{(l)}$  is minimum when  $f_1$  transmits to  $\{D_1, D_2\}$  and  $f_2$  to  $\{D_3\}$  (line 4). Then,  $f_p$ 's and the corresponding  $DSU_{f_p}^{(l)}$ 's are stored in  $F_n^{(l)}$  and  $D_n^{(l)}$  as  $F_n^{(l)} = \{\{f_1\}, \{f_2\}\}$  and  $D_n^{(l)} = \{\{D_1, D_2\}, \{D_3\}\}$  respectively (line 7). Using the packet reception status at the nodes of  $FS_n^{(l)}$  obtained in Algorithm 2, packets  $(P_2^{(l)}, P_3^{(l)})$  are present in  $FSU^{(l)}[1]$  and not in  $FS_n^{(l)} - FSU^{(l)}[1] = \{f_3\}$ . Therefore,  $f_1$  is assigned to transmit packets  $(P_2^{(l)}, P_3^{(l)})$  to  $\{D_1, D_2\}$  and  $f_2$  is assigned to transmit packets  $(P_2^{(l)}, P_3^{(l)})$  to  $\{D_3\}$  (line 8). Thus,  $IP_n^{(l)} = \{(P_2^{(l)}, P_3^{(l)}), (P_2^{(l)}, P_3^{(l)})\}$ . On following the same steps for  $FSU^{(l)}[2]$  and  $FSU^{(l)}[3]$ ,  $F_n^{(l)}$ ,  $D_n^{(l)}$ , and  $IP_n^{(l)}$  in line 15 of Algorithm 3 are updated as

$$\begin{aligned}
F_n^{(l)} &= \{\{f_2\}, \{f_3\}, \{f_2\}, \{f_1\}, \{f_1\}, \{f_1\}\} \\
D_n^{(l)} &= \{\{D_3\}, \{D_3\}, \{D_3\}, \{D_1, D_2\}, \{D_1, D_2\}, \{D_1, D_2\}\} \\
IP_n^{(l)} &= \{(P_2^{(l)}, P_3^{(l)}), (P_1^{(l)}), (P_4^{(l)}, P_5^{(l)}), (P_2^{(l)}, P_3^{(l)}), (P_1^{(l)}), (P_4^{(l)}, P_5^{(l)})\}
\end{aligned}$$

In Algorithm 4,  $n$ ,  $DS_n^{(l)}$ , and the innovative packets assigned to node  $n$  for further transmission are initialized based on the outcome of Algorithm 3. For the example being considered,  $f_1$  is assigned to act as source for communicating packets  $(P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  to  $\{D_1, D_2\}$ . Similarly,  $f_2$  and  $f_3$  are assigned to send  $(P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  and  $(P_1^{(l)})$ , respectively to  $\{D_3\}$ . Next, different cases encountered while executing Algorithm 3 and Algorithm 4 are illustrated. Note that an independent link assumption is made in the example networks.

**Case I:** Different nodes in the  $FS_n^{(l)}$  have common innovative packet but with different code vectors.

For illustration, consider the example network in Fig. 4.2 with the same outcome of Algorithm 1 and the packet reception after the first 5 transmissions in Algorithm 2. Now, assume that the innovative packet  $(P_2^{(l)} \oplus P_3^{(l)})$  transmitted by  $S^{(l)}$  is successfully received only at  $f_1$  and  $f_3$ . Then,  $S^{(l)}$  broadcasts  $(P_4^{(l)} \oplus P_5^{(l)})$ , which is successfully received at  $f_1, f_2$ , and  $f_3$ . The reception status of packets at the nodes of  $FS^{(l)}$  are  $f_1: (P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ ;  $f_2: (P_2^{(l)}, P_5^{(l)})$ ;  $f_3: (P_1^{(l)}, P_5^{(l)}, P_2^{(l)} \oplus P_3^{(l)})$ . Next, Algorithm 3 is followed considering changes in results of Algorithm 2 for the network in Fig. 4.2. The minimum multicasting cost from  $FSU^{(l)} [2]$  to  $DS^{(l)}$  is achieved when  $f_1$  transmits packets to  $\{D_1, D_2\}$  and  $f_3$  to  $\{D_3\}$ . Now,  $f_1$  has  $(P_3^{(l)})$  and  $f_3$  has  $(P_2^{(l)} \oplus P_3^{(l)})$ , which are common innovative packet at  $f_1$  and  $f_3$ , but with different code vectors, i.e.,  $(0, 0, 1, 0, 0)$  and  $(0, 1, 1, 0, 0)$ , respectively. Therefore,  $\{D_1, D_2\}$  receives  $(P_1^{(l)}, P_3^{(l)})$  from  $f_1$  while  $\{D_3\}$  receives  $(P_1^{(l)}, P_2^{(l)} \oplus P_3^{(l)})$  from  $f_3$ . Later,  $\{D_3\}$  receives  $(P_2^{(l)})$  from  $f_2$ , and thus, can extract  $(P_3^{(l)})$  by XORing  $(P_2^{(l)} \oplus P_3^{(l)})$  with  $(P_2^{(l)})$ . Therefore,  $F_n^{(l)}$ ,  $D_n^{(l)}$ , and  $IP_n^{(l)}$  on sorting are

$$\begin{aligned}
F_n^{(l)} &= \{\{f_2\}, \{f_3\}, \{f_2\}, \{f_1\}, \{f_1\}, \{f_1\}\} \\
D_n^{(l)} &= \{\{D_3\}, \{D_3\}, \{D_3\}, \{D_1, D_2\}, \{D_1, D_2\}, \{D_1, D_2\}\} \\
IP_n^{(l)} &= \{(P_2^{(l)}), (P_1^{(l)}, P_2^{(l)} \oplus P_3^{(l)}), (P_4^{(l)}, P_5^{(l)}), (P_2^{(l)}), (P_1^{(l)}, P_3^{(l)}), (P_4^{(l)}, P_5^{(l)})\}
\end{aligned}$$

In Algorithm 4, nodes of  $FS^{(l)}$  are assigned as source to communicate the corresponding batches to the destined subset of  $DS^{(l)}$ .

**Case II:** Distinct packets of a batch are assigned to the same  $FS_n^{(l)}$  node which are destined for different destination subsets

Consider the example network shown in Fig. 4.3, where  $S^{(l)} = 0$ ,  $DS^{(l)} = \{6, 7, 8\}$ , and  $FS^{(l)} = \{1, 2\}$ . A batch  $B^{(l)} = (P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  is to be reliably multicast from  $S^{(l)}$  to  $DS^{(l)}$ . Using Algorithm 1, the subset of  $FS^{(l)}$  which must have  $N^{(l)}$  innovative packets is identified as  $ENE^{(l)} = \{1\}$ . Let packets received at the nodes of  $FS^{(l)}$  after completion of Algorithm 2 be 1:  $(P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  and 2:  $(P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ . For Algorithm 3,  $FSU^{(l)} = \{\{1\}, \{1, 2\}\}$ . Using the outcomes of Algorithms 1 and 2, the outcome of Algorithm 3 i.e.  $F_n^{(l)}$ ,  $D_n^{(l)}$ , and  $IP_n^{(l)}$  on sorting are

$$\begin{aligned}
F_n^{(l)} &= \{\{2\}, \{1\}, \{1\}\} \\
D_n^{(l)} &= \{\{8\}, \{6, 7\}, \{6, 7, 8\}\} \\
IP_n^{(l)} &= \{(P_3^{(l)}, P_4^{(l)}, P_5^{(l)}), (P_3^{(l)}, P_4^{(l)}, P_5^{(l)}), (P_1^{(l)}, P_2^{(l)})\}
\end{aligned}$$

For  $n = \{1\}$  in Algorithm 4 (line 27),  $n = \{1\}$ ,  $DS_n^{(l)} = \{6, 7\}$ ,  $IP_n^{(l)} = (P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ ,  $N_n^{(l)} = 5$  and  $n = \{1\}$ ,  $DS_n^{(l)} = \{8\}$ ,  $IP_n^{(l)} = (P_1^{(l)}, P_2^{(l)})$ ,  $N_n^{(l)} = 2$ . That is, node 1 is assigned to send packets  $(P_1^{(l)}, P_2^{(l)}, P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  to  $\{6, 7\}$  and  $(P_1^{(l)}, P_2^{(l)})$  to  $\{8\}$ . Similarly, node 2 is assigned to send packets  $(P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  to  $\{8\}$ . For  $n = \{1\}$ , the outcome of Algorithm 1 is  $ENE^{(l)} = \{\{5\}, \{3, 4\}\}$ . On executing

Algorithm 2, let us assume that the packet reception status are  
 3:  $(P_1^{(l)}, P_2^{(l)}, P_3^{(l)})$ ; 4:  $(P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$ ; 5:  $(P_1^{(l)}, P_2^{(l)})$ .

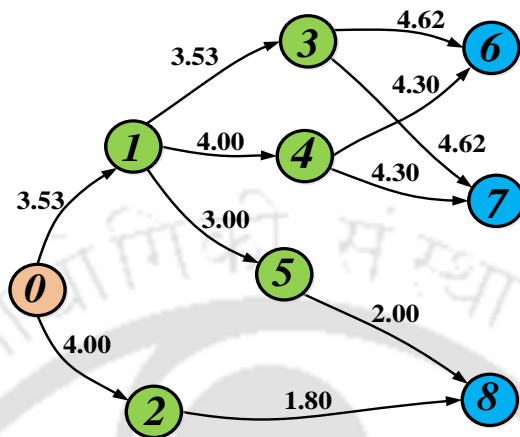


Figure 4.3: Example network 2: Multicasting

The traffic distribution is performed using Algorithm 3 with  $FSU^{(l)} = \{\{3, 5\}, \{4, 5\}, \{3, 4, 5\}\}$ .  $F_n^{(l)}$ ,  $D_n^{(l)}$ , and  $IP_n^{(l)}$  are  $F_n^{(l)} = \{\{3\}, \{5\}\}$ ,  $D_n^{(l)} = \{\{6, 7\}, \{8\}\}$ , and  $IP_n^{(l)} = \{(P_1^{(l)}, P_2^{(l)}), (P_1^{(l)}, P_2^{(l)})\}$ . It is observed that,  $(P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  are not assigned to any of the forwarding nodes. Therefore, in line 13 of Algorithm 3, a subset of  $FS_n^{(l)}$  with the unassigned packets is identified. These packets are multicast to the corresponding destinations with the least multicasting cost. In this case, node  $n = 1$  communicates the unassigned packets  $(P_3^{(l)}, P_4^{(l)}, P_5^{(l)})$  to  $DS_n^{(l)} = \{6, 7\}$  with the help of  $FS_n^{(l)} = \{3, 4\}$ . Since,  $(P_3)$  is present at node 3 and node 4, and  $C_{4, \{6,7\}}^{(l),i}$  is the least multicasting cost, node 4 is assigned to transmit  $(P_3^{(l)})$ .  $(P_4^{(l)})$  and  $(P_5^{(l)})$  are only available at node 4; therefore node 4 is assigned to transmit packets  $(P_4^{(l)}, P_5^{(l)})$ .  $F_n^{(l)}$ ,  $D_n^{(l)}$ , and  $IP_n^{(l)}$  are updated and sorted as

$$\begin{aligned}
F_n^{(l)} &= \{\{5\}, \{3\}, \{4\}\} \\
D_n^{(l)} &= \{\{8\}, \{6, 7\}, \{6, 7\}\} \\
IP_n^{(l)} &= \{(P_1^{(l)}, P_2^{(l)}), (P_1^{(l)}, P_2^{(l)}), (P_3^{(l)}, P_4^{(l)}, P_5^{(l)})\}
\end{aligned}$$

#### 4.4 Simulation Results and Complexity Analysis

In this section, simulation results and complexity analysis for the proposed NC-MLCAR scheme are presented. The multicasting cost of the scheme is compared with the other state-of-the-art schemes like MORE [4], Pacifier [7], and MLCAR [5] for both independent and correlated link networks.

##### 4.4.1 Simulation Results

An event driven simulator is designed for MORE, Pacifier, MLCAR, and NC-MLCAR in the C++ programming to obtain their multicasting costs. Each protocol is simulated for 10 randomly generated topologies with 20 static nodes. A source node  $S^{(l)}$  and destination set  $DS^{(l)}$  of size 4 are randomly selected for each topology. The probabilities of successful packet transmissions  $p_s$  on the different links of the network are randomly generated in the range  $\{0.15, 0.85\}$ . In correlated link networks, the probability of correlation between links of the network is dependent on  $p_s$ . Let  $\xi = \min\{p_{n,1}, \dots, p_{n,H}\}$  for any node  $n$  of the network, where  $p_{n,h}$ ,  $h = \{1, \dots, H\}$  is  $p_s$  for the link between  $n$  and  $h$ , and  $H$  is size of  $NS_n$ . The impact of link correlation on the performance gain is presented for three cases: (i) low link correlation,  $p_c = 10\%$  of  $\xi$ , (ii) moderate link correlation,  $p_c = 50\%$  of  $\xi$ , and (iii) high link correlation,  $p_c = 90\%$  of  $\xi$ , where  $p_c$  is the probability of correlation among the possible combinations of links from  $n$  to the respective nodes of  $NS_n$ .

MORE eliminates the forwarders with less than 10% of the batch transmission. This pruning threshold disconnects some receivers, which may obstruct in initiating the communication for the next batch by the source. Pacifier reduces the pruning threshold from 0.10 to 0.01 so that the receivers are not disconnected. The best average throughput in Pacifier is obtained by selecting the parameter value  $knob = 1$  [7]. The same pruning threshold and  $knob$  value are used while implementing MORE and Pacifier. The size of  $N^{(l)}$  is assumed to be 32, which is the same as in [4, 7]. The data file at the source node is divided into  $10^5$  batches with  $N^{(l)}$  packets in each batch.

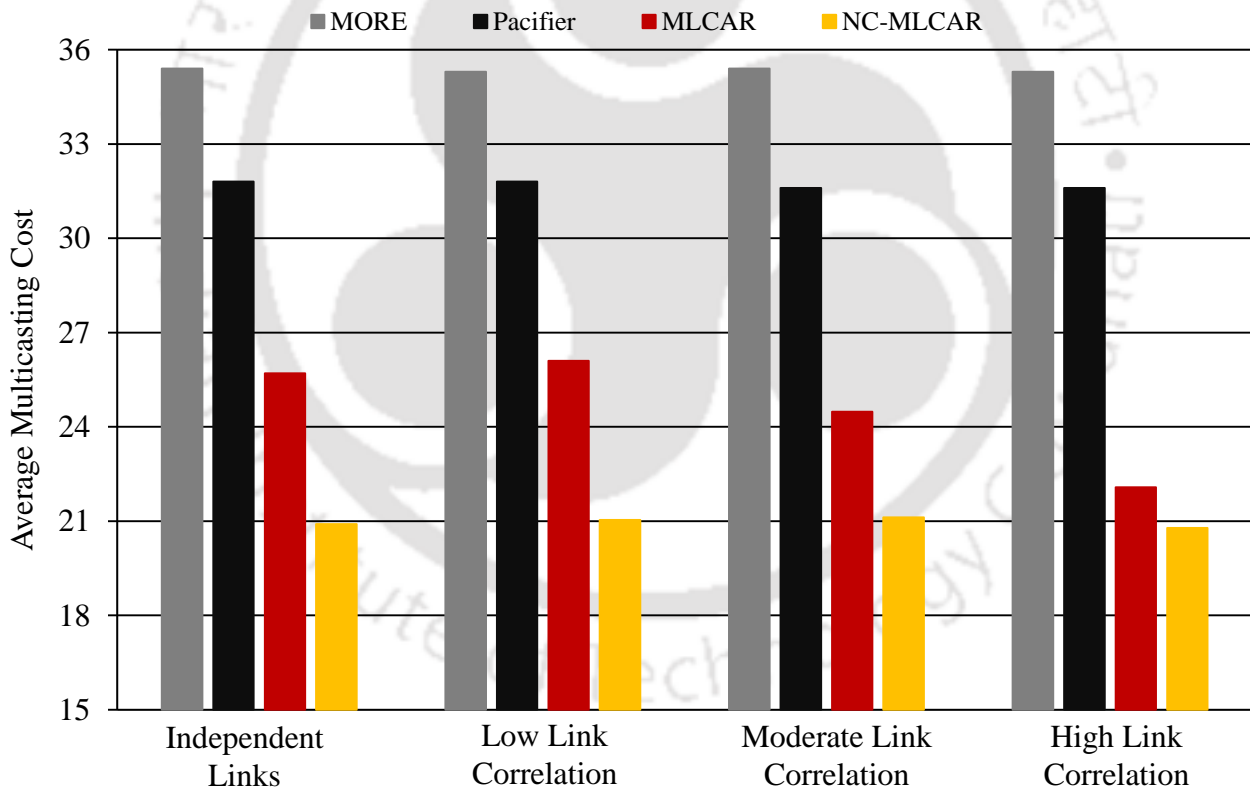


Figure 4.4: Average multicasting cost of different schemes for independent and correlated links.

The multicasting cost is defined as the ETXs required to reliably send packets from  $S^{(l)}$  to  $DS^{(l)}$ . In Fig. 4.4, the average multicasting costs of MORE, Pacifier, MLCAR, and NC-MLCAR schemes are plotted for both independent and correlated link networks. For

networks with correlated links, three cases of low (10%), moderate (50%) and high (90%) correlation are considered, as was defined earlier. It is observed that the average multicasting cost of the proposed NC-MLCAR scheme is significantly lower than that of the MORE, Pacifier, and MLCAR protocols. Further, the average multicasting cost of MORE, Pacifier, and NC-MLCAR is almost constant on varying link correlation. However, the average multicasting cost of MLCAR reduces with an increase in the network links correlation. When the correlation between the links is high, the packets received at nodes are almost identical. The decrement in the cost of NC-MLCAR over MLCAR is around 5 with a performance gain of 19% for independent links, as shown in Fig. 4.4. For correlated links, it varies with the correlation as 5 (Case 1), 3.40 (Case 2), and 1.5 (Case 3) with performance gain of 19.5%, 14%, and 6%, respectively.

#### 4.4.2 Complexity Analysis

Complexity analysis for NC-MLCAR involves identifying the complexity of route cost calculations and packet transmissions. The complexity incurred in route cost calculations using the Bellman-Ford type algorithm to reach from all possible combinations of neighboring nodes  $NS_n$  to subsets of  $DS_n^{(l)}$  is  $O(2^{(|NS_n|+|DS_n^{(l)}|)})$ . Algorithms 1 - 4 discussed above are used for packet transmission. In Algorithm 1, the subsets of  $FS_n^{(l)}$  reach each node of  $DS_n^{(l)}$  with complexity  $O(|FS_n^{(l)}||DS_n^{(l)}|)$ . The subsets of  $FS_n^{(l)}$  with  $N_n^{(l)}$  innovative packets  $ENE^{(l)}$  can be identified with complexity of  $O(|M_n^{(l)}|^2)$ . Thus, the overall complexity of Algorithm 1 is bounded by  $O(|FS_n^{(l)}||DS_n^{(l)}|+|M_n^{(l)}|^2)$ . Next, in Algorithm 2, coded packets that are innovative for the maximum number of forwarding nodes are identified. This requires packet reception status of  $N_n^{(l)}$  innovative packets from all the nodes of  $FS_n^{(l)}$  resulting in complexity of  $O(|N_n^{(l)}||FS_n^{(l)}|)$ . In Algorithm 3, a node searches all

possible combinations of  $FSU^{(l)}$  reaching the subsets of  $DS_n^{(l)}$  such that the overall multicasting cost is minimum.  $FS_n^{(l)}$  is the largest  $FSU^{(l)}$  therefore the complexity of Algorithm 3 is bounded by  $O(2^{(|FS_n^{(l)}|+|DS_n^{(l)}|)})$ . Algorithm 4 assigns nodes of  $F_n^{(l)}$  as new  $n$  and the corresponding destination set present in  $D_n^{(l)}$  as  $DS_n^{(l)}$ . The innovative packets present in the node  $n$  are assigned to  $IP_n^{(l)}$ . This incurs a complexity of  $O(|FS_n^{(l)}|/(|F_n^{(l)}|+|D_n^{(l)}|))$  in Algorithm 4.

Table 4.1: Comparison of complexity analysis of MLCAR and NC-MLCAR

	Algorithm complexity	
	MLCAR	NC-MLCAR
Route cost calculation	$O(2^{( NS_n + DS_n^{(l)} )})$	$O(2^{( NS_n + DS_n^{(l)} )})$
Packet transmission	$O(2^{( FS_n^{(l)} + DS_n^{(l)} )})$	$O(2^{( FS_n^{(l)} + DS_n^{(l)} )})+O( FS_n^{(l)} / DS_n^{(l)} + DS_n^{(l)} ^2)$ $+O( N_n^{(l)} / FS_n^{(l)} )+O( FS_n^{(l)} /( F_n^{(l)} + D_n^{(l)} ))$

The MLCAR protocol is a special case of NC-MLCAR protocol when batch size ( $N_n^{(l)}$ ) is one. The extra computational complexity of NC-MLCAR incurs because of NC. The computational complexity of NC increases with the batch size. In order to form an innovative packet, packet reception status at the nodes of  $FS_n^{(l)}$  is required for all the packets in a batch. If the batch size is large, encoding and decoding is performed using more number of packets at the senders and receivers end, respectively. For simulation, the batch size is assumed to be 32, which is same as in [4, 7]. As network size grows in size,  $|NS_n|$ ,  $|DS_n^{(l)}|$ , and  $|FS_n^{(l)}|$  increase which in turn makes the system more complex. The computational complexity of

route cost calculations and packet transmission is mainly dominated by the exponential term  $O(2^{(|NS_n|+|DS_n^{(1)}|)})$  and  $O(2^{(|FS_n^{(1)}|+|DS_n^{(1)}|)})$  respectively (Table 4.1.). Therefore, MLCAR and NC-MLCAR protocols are not suitable for the bigger networks. The simulator designed for each protocol can handle up to 50 network nodes with a maximum forwarder and destination set size of 16.

#### 4.5 Summary

This chapter presents a novel NC assisted packet forwarding strategy for wireless networks, where a source node multicasts packet to the destination set with the help of a forwarder set selected using MLCAR protocol [5]. The scheme applies to both independent and correlated link networks. The main attributes of the proposed scheme are (i) it tracks the packet status of the forwarding nodes through ACK/NACK to ensure that all the innovative packets are forwarded to the successive nodes to ultimately reach the destination set, (ii) combine packets that are innovative for the maximum number of selected receiving nodes, and (iii) distribute packets among the forwarding nodes judiciously, so that the cost to reach the destination set is minimum. Simulation results show the advantages of the proposed NC-MLCAR scheme over the existing NC assisted multicasting approaches such as MORE [4] and Pacifier [7]. The results are also compared with the corresponding MLCAR approach. A significant reduction in multicasting cost for the proposed NC-MLCAR scheme is observed. In a practical scenario, links are not independent, so simulation results considering link correlations are also presented for the different schemes. The multicasting cost of the NC-MLCAR scheme is almost constant on varying link correlation.



## Chapter 5

# Network Coding Assisted Reliable Multi-Source Multicasting in Multi-Hop Wireless Networks

The previous chapter discussed a NC assisted multicasting protocol for independent and correlated link networks. A source node transmits data to multiple destinations present in a multi-hop wireless network. In this chapter, a NC Assisted Reliable Multi-Source Multicasting (NCRMM) scheme is proposed for a multi-hop wireless network. The scheme decreases the average number of packet transmissions in a network, which reduces resource expenditure and traffic congestion problems. The extensive simulation results show that the cost of multicasting packets in NCRMM scheme are significantly less than that of existing multicasting schemes for both independent and correlated link scenarios.

### 5.1 Introduction

Wireless broadcasting facilitates the simultaneous reception of packets at multiple nodes, which would provide an opportunity to perform NC where a single coded packet can be beneficial for multiple terminals. NC is widely known to improve the overall network throughput and delay performance. Intra-session NC mixes packets of the same session, whereas inter-session NC combines packets of different sessions passing through the intermediate nodes. The NCRMM scheme uses the MLCAR algorithm [5] to select the forwarding nodes involved in packet transmissions from multiple source nodes to the corresponding destination sets. At each intermediate node, a CW is defined, which stores packets received at the node from the same and/or different sessions. The packets are judiciously combined using intra-session and inter-session NC to ensure efficient distribution of the packets among the forwarding nodes. This is done while ensuring that packets

transmitted from each source node is reliably received at all nodes of the corresponding destination set. The results and complexity analysis of the NCRMM scheme are presented and compared with the corresponding MLCAR and NC-MLCAR schemes.

The rest of the chapter is organized as follows. In Section 5.2, the implementation details of NCRMM are presented. Section 5.3 presents the proposed algorithm to transmit packets efficiently from multiple source nodes to their respective destination sets. In Section 5.4, simulation results are presented. Section 5.5 presents the complexity analysis of the NCRMM scheme. Finally, Section 5.6 concludes the chapter.

## 5.2 Implementation Details: NCRMM

Multi-source multicasting over a stationary multi-hop wireless mesh network is considered assuming  $m_s$  to be the number of multicasting sessions in a network where each session has its source  $S^{(q)}$  and the destination set  $DS^{(q)}$  of size  $M^{(q)}$  for  $q = n_m$  and  $n_m = (1, \dots, m_s)$ . The data file in each multicasting session is sent as batches. A network node can act as the source node of one session and the destination node for another session. Moreover, a node can be present in the destination set of more than one source. It is assumed that a source node does not act as an intermediate node of the network. Further, a network node transmits its packet whenever the MAC layer permits. As considered in [78], receivers snoop transmissions in promiscuous mode. The overheard packets are stored in a network node queue for a limited time-period  $T_s$ , which is assumed to be sufficient to perform NC. Packets transmitted at a network node can be native or intra-session/inter-session coded based on the packet reception status at the neighboring node(s). A coded packet is formed such that it would be an innovative packet for a maximum number of nodes. The successful and

unsuccessful packet reception status is reported to the transmitter through ACKs/NACKs. ACKs/NACKs are considered instantaneous and error-free for the sake of simplicity.

In the proposed scheme, the multicasting route with the least cost is obtained using the MLCAR protocol [5]. For the  $q^{\text{th}}$  session, forwarder sets  $FS^{(q)}$  are determined to reach from source node  $S^{(q)}$  to all possible subsets of the destination set  $DS^{(q)}$ . After multicasting cost calculation for all sessions, data transmission is initiated from the source nodes to the respective destination sets. A source node  $S^{(q)}$  of the  $q^{\text{th}}$  session transmits its data file as a batch of  $N^{(q)}$  packets. Let  $B^{(q)}$  be a batch for  $q^{\text{th}}$  session consisting of  $N^{(q)}$  packets, represented as  $B^{(q)} = (P_1^{(q)}, \dots, P_{N^{(q)}}^{(q)})$ . The packets in a batch of the  $q^{\text{th}}$  session are stored in the forwarding node queue after the packet forwarding decision is made based on the MLCAR cost. Once the packet forwarding decision is executed, the source erases the old batch and stores a new batch. A CW is defined for each forwarding node queue, which decides the maximum number of packets a node can first send using NC. Source and destination nodes do not have a CW, and source nodes follow batch-wise transmission using intra-session NC. In a CW, packets can be stored from the same session and different multicasting sessions, which provides an opportunity to perform intra-session and inter-session NC. Moreover, packets held in a CW on distribution are either native or intra-session coded packets. The forwarding nodes do not wait for the CW to be full; instead, they transmit native/network-coded packets whenever the MAC layer permits. If the CW is not empty and packets are already assigned to nodes for transmission, then they are stored in the node's queue. Once all the packets belonging to CW at the node are transmitted to the respective forwarding nodes, the node shifts to a new set of packets in its queue and flushes out the old ones. If the packet received at a node is not intended for it, then the packet is stored in the receiver's queue for decoding but is not part of its CW.

A binary NC scheme where the coding coefficients are either 0 or 1 is followed. A network-coded packet formed on combining  $P_c$  packets can be decoded if the receiver has  $P_c - 1$  of the  $P_c$  packets in its queue. Therefore, a packet is said to be innovative if it is linearly independent of the packets already stored at the node. A destination of  $q^{\text{th}}$  session can retrieve the batch once it receives all  $N^{(q)}$  innovative packets of the batch. In the proposed scheme,  $FS^{(q)}$  is selected among the neighboring nodes, so that the cost to  $DS^{(q)}$  is minimum. Network link cost metric  $ETX$  is defined as  $1/p_s$ , where  $p_s$  is the probability that packets are successfully transmitted over the link. The  $p_s$  values are assumed to be the same for bidirectional links in the network. The cost of multicasting a packet from  $S^{(q)}$  to  $DS^{(q)}$  over a network with independent links is given by (5.1)

$$C_{S^{(q)}, DS^{(q)}}^{(q),i}(f_x^{(q),i}) = \frac{P_F + \sum_{\forall f_a^{(q),i} \subseteq f_x^{(q),i}} \beta_{f_a^{(q),i}}^{(q),i} \left( 1 + C_{f_a^{(q),i}, DS^{(q)}, f_a^{(q),i}}^{(q),i} + C_{S^{(q)}, DS^{(q)}, f_y^{(q),i}}^{(q),i} \right)}{1 - P_F} \quad (5.1)$$

The multicasting cost of communicating a packet from  $S^{(q)}$  to  $DS^{(q)}$  over a network with correlated link environment is given by (5.2)

$$C_{S^{(q)}, DS^{(q)}}^{(q),c}(f_x^{(q),c}) = \frac{1 + \sum_{\forall f_a^{(q),c} \subseteq f_x^{(q),c}, f_a^{(q),c} \neq \phi} \beta_{f_a^{(q),c}}^{(q),c} \left( C_{f_a^{(q),c}, DS^{(q)}, f_a^{(q),c}}^{(q),c} + C_{S^{(q)}, DS^{(q)}, f_y^{(q),c}}^{(q),c} \right)}{\sum_{k=1}^{|f_x^{(q),c}|} (-1)^{k-1} P(f^k)} \quad (5.2)$$

The selection of  $FS^{(q)}$  for each  $S^{(q)}$  is done using the MLCAR cost defined in (5.1) and (5.2), for independent and correlated links, respectively. The detailed procedure for MLCAR cost calculation and  $FS^{(q)}$  selection for independent links is described in [5]. Chapter 3

describes the MLCAR cost calculation and  $FS^{(q)}$  selection for correlated links. For the sake of simplicity, the index  $i$  which represents independent links and  $c$  representing correlated link networks are omitted in the subsequent sections. The signaling overhead required for reliable packet transmission in a multi-source multicasting environment is the same as that for the MAC layer multicast protocol in [88]. It includes positive/negative acknowledgment (ACKs/NACKs),  $RTS$  (Ready to Send), and  $CTS$  (Clear to Send) control notifications communicated by network nodes. For transmitting  $i^{\text{th}}$  packet from network node  $n$ , the overhead packets needed are  $n_i = RTS + V_E(CTS + ACK/NACK)$ , where  $V_E$  is the size of

---

**Algorithm 1:** Finding subsets of neighboring nodes of  $n$  which must have innovative packets

---

**Initial Conditions:**  $FST_d^{(q)} = \phi$  for  $d \in DS_n^{(q)}$ ;  $AFST^{(q)}[1] = \dots = AFST^{(q)}[|AFST^{(q)}|] = \phi$ ;

$ENE^{(q)}[1], \dots, ENE^{(q)}[|ENE^{(q)}|] = \phi$ ;  $EFSN[1] = \dots = EFSN[|EFSN|] = \phi$ ;  $q \in n_m$ ;

**Input:**  $n$ ;  $FS_n^{(q)}$ ;  $DS_n^{(q)}$ ;  $C_{a,b}^{(q),w}$ ;  $a \in S^{(q)}$ ,  $b \in DS^{(q)}$ ,  $w \in \{i, c\}$ ;  $q \in n_m$

**Output:**  $ENE^{(q)}$ ;  $K^{(q)}$ ;  $q \in n_m$ ;  $EFSN$ ;

1. **for** each  $q \in n_m$  **do**
  2.     **for** each  $d \in DS_n^{(q)}$  **do**
  3.         **for** each  $f \in FS_n^{(q)}$  **do**
  4.             **if**  $C_{f,d}^{(q),w} < \infty$ ,  $w \in \{i, c\}$  **then**
  5.                  $FST_d^{(q)} = FST_d^{(q)} \cup f$
  6.             **end if**
  7.         **end for**
  8.     **end for**
  9.     Sort  $FST_d^{(q)}$ 's in ascending order of their size and store in  $AFST^{(q)}$
  10.    **for**  $l = 1$  to  $|DS_n^{(q)}|$  **do**
  11.         **for**  $m = (l + 1)$  to  $|DS_n^{(q)}|$  **do**
  12.             **if**  $AFST^{(q)}[l] \neq \phi$ ,  $AFST^{(q)}[m] \neq \phi$ ,  $AFST^{(q)}[l] \subseteq AFST^{(q)}[m]$  **then**
  13.                  $AFST^{(q)}[m] = \phi$
  14.             **end if**
  15.         **end for**
  16.     **end for**
  17.     Store the elements of  $AFST^{(q)}$  in  $ENE^{(q)}$  excluding all null elements
  18.      $EFSN = EFSN \cup ENE^{(q)}$
  19. **end for**
-

$EFSN$  associated with node  $n$ .

### 5.3 Proposed Algorithm: NCRMM

In this section, an efficient packet transmission strategy is presented for simultaneous communication of packets from the source nodes to the associated destination sets using NC. The proposed NCRMM scheme is presented with the help of a flowchart in Fig. 5.1. Further, detailed algorithms are presented and explained with the help of example networks. In Algorithm 1, the forwarder sets  $FS_n^{(q)}$  are identified that must receive all innovative packets associated with the  $q^{\text{th}}$  session present in CW of the network node  $n$  ( $CW_n$ ). This assists in delivering innovative packets of the session to the desired destination sets. With the help of Algorithm 2, reliable packet transmission is done using NC from CW of intermediate node  $n$  to the respective forwarding nodes obtained from Algorithm 1. The source nodes apply Algorithm 1 to find forwarding nodes which must have respective innovative packets and they do batch-wise transmission employing intra-session NC using Algorithm 2. Algorithm 3 is used to distribute judiciously the packets of  $q^{\text{th}}$  session among the nodes of  $FS_n^{(q)}$ , such that each destination belonging to the session receives  $N_n^{(q)}$  innovative packets with the least multicasting cost. After successfully distributing packets in Algorithm 3, Algorithm 4 updates  $n$ ,  $DS_n^{(q)}$ ,  $CW_n$ , and  $N_n^{(q)}$  to communicate the packets further towards the desired destination nodes.

Multi-source multicasting routing algorithm is explained using an example network with three multicasting sessions,  $m_s = 3$  and  $n_m = \{1, 2, 3\}$  as shown in Fig. 5.2. The source nodes and their associated destination sets along with intended batches containing packets are  $S^{(1)} = \{a_2\}$ ,  $DS^{(1)} = \{a_6, a_7, a_8\}$ ,  $B^{(1)} = \{P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\}$ ;  $S^{(2)} = \{a_6\}$ ,  $DS^{(2)} = \{a_1, a_2\}$ ,

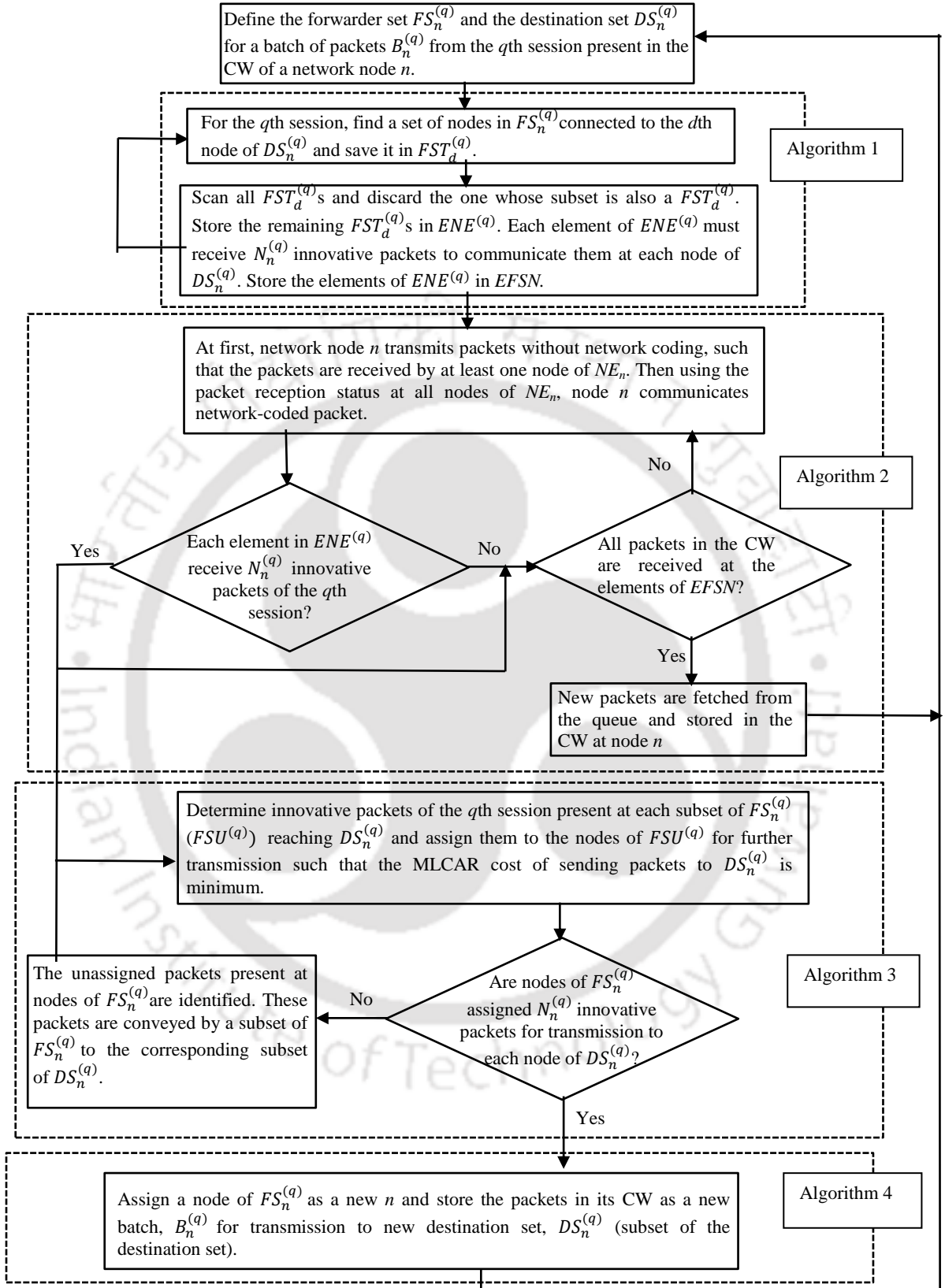


Figure 5.1: Flowchart of NCRMM

$B^{(2)} = \{P_1^{(2)}, P_2^{(2)}\}$ ; and  $S^{(3)} = \{a_7\}$ ,  $DS^{(3)} = \{a_1, a_2\}$ ,  $B^{(3)} = \{P_1^{(3)}, P_2^{(3)}\}$ . For each multicasting session, the MLCAR protocol is individually applied to determine the respective forwarder sets with least multicasting cost. The forwarder sets obtained for the network in Fig. 5.2 are  $FS^{(1)} = \{a_3, a_4, a_5\}$ ,  $FS^{(2)} = \{a_3, a_4\}$ , and  $FS^{(3)} = \{a_3, a_4\}$ .

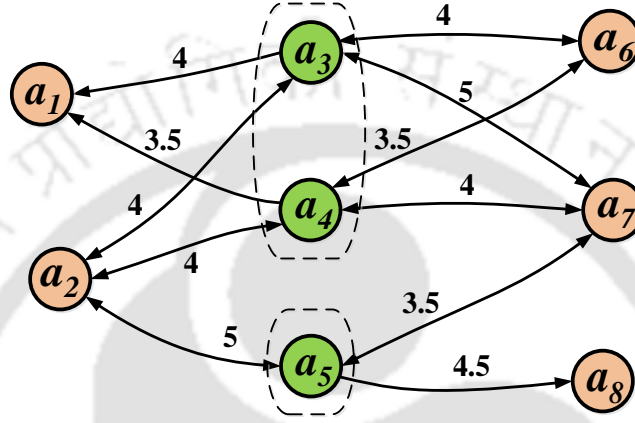


Figure 5.2: Example network 1: Multi-source multicasting

Let the MAC layer initiate packet transmission from the source nodes  $n$ ,  $n \in \{a_2, a_6, a_7\}$  to the desired destination sets. Algorithm 1 determines subsets of the forwarder set for each multicasting source  $n$ ,  $n \in \{a_2, a_6, a_7\}$ . When  $n$  is a forwarding node of the network, Algorithm 1 finds the respective forwarder sets for each session present in  $CW_n$ . Consider  $q = 1$  and  $n = \{a_2\}$ , then  $DS_{a_2}^{(1)} = \{a_6, a_7, a_8\}$  and  $FS_{a_2}^{(1)} = \{a_3, a_4, a_5\}$ . In Algorithm 1,  $FST_{a_6}^{(1)} = \{a_3, a_4\}$ ,  $FST_{a_7}^{(1)} = \{a_3, a_4, a_5\}$ , and  $FST_{a_8}^{(1)} = \{a_5\}$  (line 2 - 8). Following line 9 of Algorithm 1,  $AFST^{(1)} = \{\{a_5\}, \{a_3, a_4\}, \{a_3, a_4, a_5\}\}$ . As each element of  $AFST^{(1)}$  transmits packets to a node of  $DS_{a_2}^{(1)}$ , therefore every element of  $AFST^{(1)}$  must have innovative packets associated with session 1. Now,  $AFST^{(1)}[1] = \{a_5\}$  and  $AFST^{(1)}[2] = \{a_3, a_4\}$  are subsets of  $AFST^{(1)}[3] = \{a_3, a_4, a_5\}$ , therefore  $AFST^{(1)}[1]$

and  $AFST^{(l)}[2]$  are retained and the set with the larger set size  $AFST^{(l)}[3]$  is eliminated. The remaining elements are stored in  $ENE^{(l)}$  (line 10 - 17). Since node  $n = \{a_2\}$  has the packets of single multicasting session, therefore  $EFSN = ENE^{(l)} = \{\{a_5\}, \{a_3, a_4\}\}$  in line 18 of Algorithm 1.

---



---

**Algorithm 2:** Packet Transmission Strategy

---

**Input:**  $IPCW_n$ ;  $NE_n$ ;  $ENE^{(q)}$ ;  $q \in n_m$ ;

**Output:** Packet reception status at nodes of  $NE_n$ ;

1. **for** each packet  $p_t \in IPCW_n$  **do**
  2.     **if**  $p_t$  is received by at least one node of  $NE_n$  **then**
  3.         **continue**
  4.     **else**
  5.         node  $n$  transmits  $p_t$
  6.     **end if**
  7. **end for**
  8. **for** each  $q \in n_m$  **do**
  9.     **if** all packets in a batch for the  $q^{\text{th}}$  session are received at each elements of  $ENE^{(q)}$  **then**
  10.         **go to** step 1 of Algorithm 3
  11.     **else**
  12.         using packet reception status at the nodes in  $NE_n$ , node  $n$  transmits native/intra-session/inter-session network-coded packet and **go to** step 8 of Algorithm 2
  13.     **end if**
  14. **end for**
- 

Next, Algorithm 2 presents the packet transmission strategy. Initially, node  $n$  transmits all the packets present in CW without NC such that each packet is received by at least one node of  $NE_n$  (lines 1 - 7). Then, network-coded packets are transmitted based on the packet reception status at the nodes in  $NE_n$ . Since  $n = \{a_2\}$  is the source node, packets of the batch are transmitted from node  $a_2$ . Let the packet reception status at the nodes in  $NE_{a_2}$  be  $a_3 = \{P_2^{(l)}\}$ ,  $a_4 = \{P_2^{(l)}, P_3^{(l)}\}$ , and  $a_5 = \{P_1^{(l)}, P_3^{(l)}\}$  on executing lines 1 - 7 of

---

**Algorithm 3:** Traffic Distribution of the  $q^{\text{th}}$  session

---

**Initial Conditions:**  $F_n^{(q)}[1] = F_n^{(q)}[2] = \dots = \phi$ ;  $D_n^{(q)}[1] = D_n^{(q)}[2] = \dots = \phi$ ;

$IP_n^{(q)}[1] = IP_n^{(q)}[2] = \dots = \phi$ ;  $q \in n_m$ ;

**Input Conditions:** Packet reception status at nodes of  $FS_n^{(q)}$ ;  $N^{(q)}$ ;  $FS_n^{(q)}$ ;  $DS_n^{(q)}$ ;

**Output:**  $F_n^{(q)}$ ;  $D_n^{(q)}$ ;  $IP_n^{(q)}$ ;

1. **for**  $l = 1$  to  $|FSU^{(q)}|$  **do**
  2.     **for** each  $f_p \in FSU^{(q)}[l]$  **do**
  3.         **for** each  $DSU_{f_p}^{(q)}$  **do**
  4.             Find nodes  $f_p$ 's and the corresponding  $DSU_{f_p}^{(q)}$ 's, such that the overall cost to reach from  $FSU^{(q)}[l]$  to  $DS_n^{(q)}$  ( $C_{FSU^{(q)}[l], DS_n^{(q)}}^{(q), w}$ ,  $w \in \{i, c\}$ ) is minimum
  5.             **end for**
  6.         **end for**
  7.             The  $f_p$ 's and the corresponding  $DSU_{f_p}^{(q)}$ 's are stored as sequence in  $F_n^{(q)}$  and  $D_n^{(q)}$ , respectively
  8.             Innovative packets of the  $q^{\text{th}}$  session successfully received at all the nodes of  $FSU^{(q)}[l]$  and not at nodes of  $FS_n^{(q)} - FSU^{(q)}[l]$  are stored against the corresponding  $F_n^{(q)}$  and  $D_n^{(q)}$  in  $IP_n^{(q)}$ . Delete the null elements of  $IP_n^{(q)}$  along with the respective element in  $F_n^{(q)}$  and  $D_n^{(q)}$ , if any
  9.         **end for**
  10.         **if** nodes of  $FS_n^{(q)}$  are assigned to transmit  $N^{(q)}$  innovative packets to each node of  $DS_n^{(q)}$  **then**
  11.             **continue**
  12.             **else**
  13.                 Determine the unassigned packets at the nodes of  $FS_n^{(q)}$  and the corresponding nodes of  $DS_n^{(q)}$  (subset of  $DS_n^{(q)}$ ) with the unassigned packets. Find the subset of  $FS_n^{(q)}$  with these unassigned packets and declare this subset of  $FS_n^{(q)}$  as a new  $FS_n^{(q)}$ . This new  $FS_n^{(q)}$  transmits the unassigned packets to the identified subset of  $DS_n^{(q)}$  (as new  $DS_n^{(q)}$ ) and thereby **go to** step 1 of Algorithm 3
  14.             **end if**
  15.         Sort elements of  $D_n^{(q)}$  in ascending order of their size and arrange  $F_n^{(q)}$  and  $IP_n^{(q)}$  based on modified index order of  $D_n^{(q)}$  on sorting
- 

Algorithm 2. Now, packets  $P_2^{(1)}$  and  $P_1^{(1)}$  are innovative for  $EFSN[1]$  and  $EFSN[2]$ , respectively. Thus, intra-session coded packet ( $P_1^{(1)} \oplus P_2^{(1)}$ ) is transmitted (line 12).

Considering that the packet is reliably received at nodes  $a_3$  and  $a_5$ , the packet reception

status at nodes is updated as  $a_3 = \{P_1^{(1)}, P_2^{(1)}\}$ ,  $a_4 = \{P_2^{(1)}, P_3^{(1)}\}$ , and  $a_5 = \{P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\}$ .

Since all the elements of  $EFSN$  have received the desired packets, Algorithm 3 (line 10) is executed.

In Algorithm 3, packets distribution based on multicasting cost is done among nodes of the forwarder set. These nodes then transmit the assigned packets to the destination set. In lines 1 - 6 of Algorithm 3,  $f_p$ 's and the corresponding  $DSU_{f_p}^{(q)}$ 's are determined for the  $q^{\text{th}}$  session such that multicasting cost  $C_{FSU^{(q)}[l], DS_n^{(q)}}^{(q),w}$  is minimum. The identified  $f_p$ 's and corresponding  $DSU_{f_p}^{(q)}$ 's are stored in  $F_n^{(q)}$  and  $D_n^{(q)}$ , respectively (line 7). Using packet reception status at nodes of  $FS_n^{(q)}$  from Algorithm 2, the packets present at all the nodes of  $FSU^{(q)}[l]$  and absent at  $FS_n^{(q)}-FSU^{(q)}[l]$  are found and stored in  $IP_n^{(q)}$  for the corresponding elements of  $F_n^{(q)}$  and  $D_n^{(q)}$ , respectively (line 8). This can be explained using the example network in Fig. 5.2. Consider that for  $n=\{a_2\}$ ,  $q=1$ ,  $FS_{a_2}^{(1)}=\{a_3, a_4, a_5\}$ , and  $FSU^{(1)}=\{\{a_3, a_5\}, \{a_4, a_5\}, \{a_3, a_4, a_5\}\}$ . In the case of  $FSU^{(1)}[3]$ , the MLCAR cost to communicate a packet from  $FSU^{(1)}[3]$  to  $DS_{a_2}^{(1)}=\{a_6, a_7, a_8\}$  is minimum when  $\{a_4\}$  transmit packet to  $\{a_6\}$  and  $\{a_5\}$  to  $\{a_7, a_8\}$ . The  $f_p$ 's and corresponding  $DSU_{f_p}^{(q)}$ 's are stored in  $F_n^{(q)}$  and  $D_n^{(q)}$ , respectively. The packets received at nodes of  $FS_n^{(q)}$  on executing Algorithm 2 are traced to determine the packets present at  $FSU^{(1)}[3]=\{a_3, a_4, a_5\}$  and absent at  $FS_{a_2}^{(1)}-FSU^{(1)}[3]=\{\phi\}$ . Using line 8 of Algorithm 3, let packet  $\{P_2^{(1)}\}$  is present at  $FSU^{(1)}[1]$  and absent at  $FS_{a_2}^{(1)}-FSU^{(1)}[1]$ . Now,  $\{a_4\}$  is assigned packet  $\{P_2^{(1)}\}$  for transmitting to  $\{a_6\}$  and  $\{a_5\}$  is assigned packet  $\{P_2^{(1)}\}$  for transmitting to  $\{a_7, a_8\}$ , and

---

**Algorithm 4: Initialization**

---

**Input Conditions:**  $F_n^{(q)}$ ;  $D_n^{(q)}$ ;  $IP_n^{(q)}$ ;  $FS_n^{(q)}$ ;**Output:**  $n$ ;  $DS_n^{(q)}$ ;  $IPCW_n$ ;  $N^{(q)}$ ;

1. **for** each  $f_u \in FS_n^{(q)}$  **do**
  2.  $i = 1; j = 1; DE_{f_u}^{(q)}[1] = DE_{f_u}^{(q)}[2] = \dots = \phi; PE_{f_u}^{(q)}[1] = PE_{f_u}^{(q)}[2] = \dots = \phi;$
  3. **for** each  $f_v \in F_n^{(q)}$  **do**
  4. **if**  $f_u == f_v$  **then**
  5. **if**  $DE_{f_u}^{(q)}[1] == \phi$  **then**
  6.  $DE_{f_u}^{(q)}[1] = DE_{f_u}^{(q)}[1] \cup D_n^{(q)}[v]; PE_{f_u}^{(q)}[1] = PE_{f_u}^{(q)}[1] \cup IP_n^{(q)}[v]$
  7. **end if**
  8. **if**  $j > 1$  **then**
  9.  $DE_{f_u}^{(q)}[j] = DE_{f_u}^{(q)}[j] \cup D_n^{(q)}[v]$
  10. **for**  $i = 1$  to  $j-1$  **do**
  11. **if**  $DE_{f_u}^{(q)}[j] \cap DE_{f_u}^{(q)}[i] = DE_{f_u}^{(q)}[i], DE_{f_u}^{(q)}[i] \neq \phi$  **then**
  12.  $PE_{f_u}^{(q)}[i] = PE_{f_u}^{(q)}[i] \cup IP_n^{(q)}[v]$
  13. **end if**
  14. **if**  $(DE_{f_u}^{(q)}[j] \cup DE_{f_u}^{(q)}[i]) - DE_{f_u}^{(q)}[i] \neq \phi, DE_{f_u}^{(q)}[i] \neq \phi$  **then**
  15.  $DE_{f_u}^{(q)}[j] = (DE_{f_u}^{(q)}[j] \cup DE_{f_u}^{(q)}[i]) - DE_{f_u}^{(q)}[i]$
  16.  $PE_{f_u}^{(q)}[j] = PE_{f_u}^{(q)}[j] \cup IP_n^{(q)}[v]$
  17. **else**
  18.  $DE_{f_u}^{(q)}[j] = \phi, PE_{f_u}^{(q)}[j] = \phi$
  19. **end if**
  20. **end for**
  21. **end if**
  22.  $j = j + 1$
  23. **end if**
  24. **end for**
  25. Remove null elements of  $DE_{f_u}^{(q)}$  and  $PE_{f_u}^{(q)}$ . Set  $t^{(q)} = |DE_{f_u}^{(q)}| = |PE_{f_u}^{(q)}|$ .
  26. **if**  $t^{(q)} > 0$  **then**
  27. Initialize:  $n = f_u, DS_n^{(q)} = \{DE_{f_u}^{(q)}[1], \dots, DE_{f_u}^{(q)}[t^{(q)}]\},$   
 $CW_n = \{CW_n \cup PE_{f_u}^{(q)}[1], \dots, CW_n \cup PE_{f_u}^{(q)}[t^{(q)}]\},$   
 $N^{(q)} = \max\{\text{size}(PE_{f_u}^{(q)}[1]), \dots, \text{size}(PE_{f_u}^{(q)}[t^{(q)}])\}$  and **go to** step 1 of Algorithm 1
  28. **else**
  29. **go to** step 1 of Algorithm 4
  30. **end if**
  31. **end for**
-

node  $\{a_3\}$  simply drops the packet  $\{P_2^{(1)}\}$  from its queue. The packet  $\{P_2^{(1)}\}$  is stored against the corresponding elements of  $F_n^{(q)}$  and  $D_n^{(q)}$  in  $IP_n^{(q)}$ . Similar steps are followed for  $FSU^{(1)}[1]$  and  $FSU^{(1)}[2]$ ,  $F_{a_2}^{(1)}$ ,  $D_{a_2}^{(1)}$ , and  $IP_{a_2}^{(1)}$  are updated (line 15 of Algorithm 3) as follows

$$\begin{aligned} F_{a_2}^{(1)} &= \{\{a_3\}, \{a_5\}, \{a_4\}, \{a_5\}, \{a_4\}, \{a_5\}\} \\ D_{a_2}^{(1)} &= \{\{a_6\}, \{a_7, a_8\}, \{a_6\}, \{a_7, a_8\}, \{a_6\}, \{a_7, a_8\}\} \\ IP_{a_2}^{(1)} &= \{(P_1^{(1)}), (P_1^{(1)}), (P_3^{(1)}), (P_3^{(1)}), (P_2^{(1)}), (P_2^{(1)})\} \end{aligned}$$

Now, the intermediate nodes  $\{a_3\}$ ,  $\{a_4\}$ , and  $\{a_5\}$  act as a source to communicate the packets further to the respective destination sets. Algorithm 4 updates information on source, destination set, and packets in CW of the source. After execution of Algorithm 4,  $n = \{a_3\}$ ,  $DS_{a_3}^{(1)} = \{a_6\}$ ,  $IPCW_{a_3} = \{P_1^{(1)}\}$ ,  $n = \{a_4\}$ ,  $DS_{a_4}^{(1)} = \{a_6\}$ ,  $IPCW_{a_4} = \{P_2^{(1)}, P_3^{(1)}\}$ , and  $n = \{a_5\}$ ,  $DS_{a_5}^{(1)} = \{a_7, a_8\}$ ,  $IPCW_{a_5} = \{P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\}$ .

Assume that after execution of the four algorithms discussed above for all three sessions, the packet reception status at CW with size four is  $a_3 = \{P_1^{(1)}, P_1^{(2)}, P_1^{(3)}\}$ ,  $a_4 = \{P_2^{(1)}, P_3^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ , and  $a_5 = \{P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\}$ . For  $n = \{a_4\}$ ,  $IPCW_{a_4} = \{P_2^{(1)}, P_3^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ , where packets  $\{P_2^{(1)}, P_3^{(1)}\}$  are intended for destination node  $\{a_6\}$  and  $\{P_2^{(2)}, P_3^{(2)}\}$  for nodes  $\{a_1, a_2\}$ . On executing Algorithm 1, the following are the outcomes for the three sessions

$$\begin{aligned} q = 1: ENE^{(1)} &= \{a_6\}; EFSN = \{a_6\} \\ q = 2: ENE^{(2)} &= \{\{a_1\}, \{a_2\}\}; EFSN = \{\{a_6\}, \{a_1\}, \{a_2\}\} \\ q = 3: ENE^{(3)} &= \{\{a_1\}, \{a_2\}\}; EFSN = \{\{a_6\}, \{a_1\}, \{a_2\}\} \end{aligned}$$

Let the packet reception status at  $NE_n = \{\{a_1\}, \{a_2\}, \{a_6\}\}$  on running line 1 - 7 of Algorithm 2 will be  $a_1 = \{P_2^{(1)}, P_2^{(2)}\}$ ,  $a_2 = \{P_3^{(1)}\}$ ,  $a_6 = \{P_2^{(1)}, P_2^{(3)}\}$ . Using packet reception at nodes in  $NE_n$ , node  $\{a_4\}$  transmits a network-coded packet which is innovative for maximum number of elements in  $EFSN$ . Node  $\{a_4\}$  can transmit a native packet  $\{P_2^{(3)}\}$  or an inter-session coded packet  $\{P_3^{(1)} \oplus P_2^{(3)}\}$ . Suppose  $\{a_4\}$  communicates  $\{P_2^{(3)}\}$  which is received at node  $\{a_1\}$  and  $\{a_2\}$ . After this, the nodes transmits the next innovative packet  $\{P_2^{(2)}\}$  which is only received at node  $\{a_6\}$ . The new packet reception status is  $a_1 = \{P_2^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ ,  $a_2 = \{P_3^{(1)}, P_2^{(3)}\}$ ,  $a_6 = \{P_2^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ . Further, an inter-session coded packet  $\{P_3^{(1)} \oplus P_2^{(2)}\}$  is transmitted, and that is received at nodes  $\{a_2\}$  and  $\{a_6\}$ . Thus, the packet reception status is updated as  $a_1 = \{P_2^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ ,  $a_2 = \{P_3^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ ,  $a_6 = \{P_2^{(1)}, P_3^{(1)}, P_2^{(2)}, P_2^{(3)}\}$ . Since the packets stored at the CW of  $\{a_4\}$  are received at the desired receivers, they are removed from the CW and new packets are stored from the queue. This process will eventually lead to packet reception from all source nodes to the respective destination sets.

The different cases which are encountered while distributing packets of a session among the forwarding nodes are presented in Chapter 4. Next, the possible cases while executing Algorithm 1 and 2 are discussed. With the help of Fig. 5.3, three cases are represented for a CW of size four.

**Case I:** No node in the elements of  $EFSN$  overlaps

On following Algorithm 1, let  $EFSN = \{\{A, B\}, \{C\}, \{D\}\}$  as depicted in Fig. 5.3(a). Assume forwarder node  $f_i$  has packets of three sessions in its CW. The packets  $\{P_1^{(1)}, P_2^{(1)}\}$  from session 1 should reach  $\{A, B\}$ , that is nodes  $\{A\}$  or  $\{B\}$  should together have both the

packets. Further, nodes  $\{C\}$  and  $\{D\}$  should receive packets  $\{P_1^{(2)}\}$  and  $\{P_1^{(3)}\}$ . Suppose the packet reception status at  $NE_{f_i}$  on following Algorithm 2 (line 1 - 7) is  $A = \{P_1^{(2)}\}$ ,  $B = \{P_2^{(1)}, P_1^{(3)}\}$ ,  $C = \{P_1^{(1)}, P_1^{(2)}\}$ , and  $D = \{P_1^{(3)}\}$ . Thus, the innovative packet communicated by node  $f_i$  is  $\{P_1^{(2)} \oplus P_1^{(3)}\}$ ; the packet is received by node  $\{D\}$  only. Using

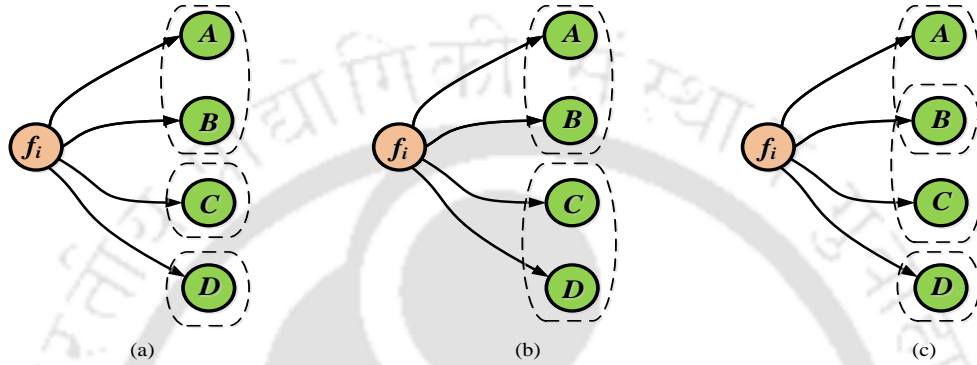


Figure 5.3: Example network 2: Multi-source multicasting

the packet reception status, consider  $\{P_1^{(1)} \oplus P_1^{(3)}\}$  is communicated and it is received at node  $\{A\}$  only. The packet reception status is updated as  $A = \{P_1^{(2)}, P_1^{(1)} \oplus P_1^{(3)}\}$ ,  $B = \{P_2^{(1)}, P_1^{(3)}\}$ ,  $C = \{P_1^{(1)}, P_1^{(2)}\}$ , and  $D = \{P_1^{(2)}, P_1^{(3)}\}$ . Next, node  $f_i$  communicates packet  $\{P_1^{(3)}\}$  as the packet is innovative for maximum number of elements in  $EFSN$ . The packet is assumed to be received at nodes  $\{A, C\}$ . Node  $\{A\}$  extracts  $\{P_1^{(1)}\}$  by XORing packets  $\{P_1^{(1)} \oplus P_1^{(3)}\}$  and  $\{P_1^{(3)}\}$ . Finally, all the desired packets are received at the neighboring nodes and the updated packet reception status is  $A = \{P_1^{(1)}, P_1^{(2)}, P_1^{(3)}\}$ ,  $B = \{P_2^{(1)}, P_1^{(3)}\}$ ,  $C = \{P_1^{(1)}, P_1^{(2)}, P_1^{(3)}\}$ , and  $D = \{P_1^{(2)}, P_1^{(3)}\}$ .

**Case II:** Elements of  $EFSN$  are a group of nodes with no common node among its elements.

Consider the network in Fig. 5.3(b) with  $EFSN = \{\{A, B\}, \{C, D\}\}$  and forwarder node  $f_i$  having packets of two sessions in its CW. Thus, together nodes  $\{A, B\}$  should have packets

of one session  $\{P_1^{(1)}, P_2^{(1)}\}$  and nodes  $\{C, D\}$  should have packets of another session  $\{P_1^{(2)}, P_2^{(2)}\}$ . On using Algorithm 2 (line 1 - 7), let the packet reception status at  $NE_{f_i}$  be  $A = \{P_2^{(1)}, P_1^{(2)}\}$ ,  $B = \{P_2^{(2)}\}$ ,  $C = \{P_1^{(1)}\}$ , and  $D = \{P_2^{(1)}, P_2^{(2)}\}$ . Using the packet reception status at the nodes of  $NE_{f_i}$ , an inter-session coded packet  $\{P_1^{(1)} \oplus P_1^{(2)}\}$  is transmitted, that is received at node  $\{B\}$ . Next, another effective innovative packet  $\{P_1^{(2)}\}$  is sent from  $f_i$  and received by nodes  $\{B, C\}$ . Node  $\{B\}$  extracts packet  $\{P_1^{(1)}\}$  by XORing packets  $\{P_1^{(2)}\}$  and  $\{P_1^{(1)} \oplus P_1^{(2)}\}$  and the modified packet reception status is  $A = \{P_2^{(1)}, P_1^{(2)}\}$ ,  $B = \{P_1^{(1)}, P_1^{(2)}, P_2^{(2)}\}$ ,  $C = \{P_1^{(1)}, P_1^{(2)}\}$ , and  $D = \{P_2^{(1)}, P_2^{(2)}\}$ .

**Case III:** Elements of EFSN are group of nodes having common nodes as its elements.

Consider the forwarder node  $f_i$  in Fig. 5.3(c) has packets of three sessions in its CW with  $EFSN = \{\{A, B\}, \{B, C\}, \{D\}\}$  on following Algorithm 1. Thus, nodes  $\{A, B\}$  together should have packets of first session  $\{P_1^{(1)}, P_2^{(1)}\}$  and nodes  $\{B, C\}$  together should have packet of second session  $\{P_1^{(2)}\}$ , and node  $\{D\}$  requires packet of third session  $\{P_1^{(3)}\}$ . Next, Algorithm 2 (line 1 - 7) is pursued and the packet reception status at the nodes of  $NE_{f_i}$  is noted as  $A = \{P_1^{(2)}\}$ ,  $B = \{P_1^{(3)}\}$ ,  $C = \{P_2^{(1)}\}$ , and  $D = \{P_1^{(1)}\}$ . Further, an inter-session coded packet  $\{P_2^{(1)} \oplus P_1^{(2)}\}$  is transmitted and it is received by nodes  $\{B, D\}$ . Next, another effective innovative packet  $\{P_1^{(2)}\}$  is send and the packet is received at nodes  $\{B, C, D\}$ . Node  $B$  extracts packet  $\{P_2^{(1)}\}$  on XORing packets  $\{P_1^{(2)}\}$  and  $\{P_2^{(1)} \oplus P_1^{(2)}\}$ . Then, packet  $\{P_1^{(1)} \oplus P_1^{(3)}\}$  is communicated and it is received at all the neighboring nodes  $\{A, B, C, D\}$ .

The updated packet reception status is  $A = \{P_1^{(1)} \oplus P_1^{(3)}, P_1^{(2)}\}$ ,  $B = \{P_1^{(1)}, P_2^{(1)}, P_1^{(2)}, P_1^{(3)}\}$ ,  $C = \{P_2^{(1)}, P_1^{(2)}, P_1^{(1)} \oplus P_1^{(3)}\}$ , and  $D = \{P_1^{(1)}, P_2^{(1)}, P_1^{(2)}, P_1^{(3)}\}$ .

In NCRMM, the first-hop forwarding nodes ( $f_u$ 's) of the actual source ( $S^{(q)}$ , 'q' represents multicasting session) act as a source node to forward packets present in its CW further over the network. Algorithm 3 in chapter 5 is employed to distribute packets at node  $S^{(q)}$  to  $f_u$ 's. These forwarding nodes ( $f_u$ 's) may act as forwarding nodes for other source nodes of the network. Therefore, the CW of  $f_u$  might have packets of different multicasting sessions. Node  $f_u$  acts as a source and executes Algorithm 1 to obtain set of forwarding nodes that must receive packets present in the CW of  $f_u$ . Algorithm 2 is executed at  $f_u$  to efficiently transmit packets present in its CW to the nodes obtained by Algorithm 1. After that, Algorithm 3 is run to distribute these packets among the forwarding nodes of  $f_u$  efficiently. Lastly, Algorithm 4 assigns the forwarding nodes of  $f_u$  as a new source to further transmit packets over the network. This process continues until packets of all the multicasting sessions are received at their intended destination nodes.

#### 5.4 Simulation Results and Complexity Analysis

In section 5.4.1, the simulation results of the NCRMM scheme are shown for packet-wise data transmissions with  $m_s = 2$ . The batch size of each multicasting session is considered to be one i.e.  $N^{(1)} = 1$  and  $N^{(2)} = 1$ . This special case of NCRMM is named Two-Source Multicasting using NC (TSMNC). Section 5.4.2 presents simulation results of the NCRMM scheme and compares them with the corresponding MLCAR and NC-MLCAR schemes for both independent and correlated link networks. Additionally, the complexity of the NCRMM

scheme is analyzed and compared with that of the MLCAR and NC-MLCAR schemes in Section 5.4.3.

#### 5.4.1 Simulation Results of TSMNC

In this subsection, the multicasting cost of the TSMNC scheme is compared with that of the MLCAR protocol [5]. An event-driven simulator is designed in the C++ programming language to obtain the results. To get more persuasive results, each protocol is simulated for 10 randomly generated topologies with 20 static nodes. Source node  $S^{(1)}$ ,  $S^{(2)}$  and the corresponding destination set  $DS^{(1)}$ ,  $DS^{(2)}$ , respectively are randomly selected for each topology where  $|DS^{(1)}| = |DS^{(2)}| = 4$ . The probabilities of successful packet transmissions  $p_s$  for different links of the network are randomly generated in the range  $\{0.15, 0.85\}$ . At the source node, the data file is divided into  $10^4$  packets. The ETXs required to reliably communicate packets from  $\{S^{(1)}, S^{(2)}\}$  to  $\{DS^{(1)}, DS^{(2)}\}$  is the multicasting cost. In Fig. 5.4, the average multicasting cost of the 10 random networks is plotted for TSMNC and MLCAR schemes. It is observed that the multicasting cost of the TSMNC scheme is significantly smaller than that for the MLCAR protocol. As CW size increases, opportunities for combining packets grow, which reduces multicasting cost but increases the delay of some packets. The reduction in the average cost of TSMNC over MLCAR is around 9.10 when CW size is 16 (performance gain is 17.84%) and 12.88 when CW size is 32 (performance gain is 25.30%).

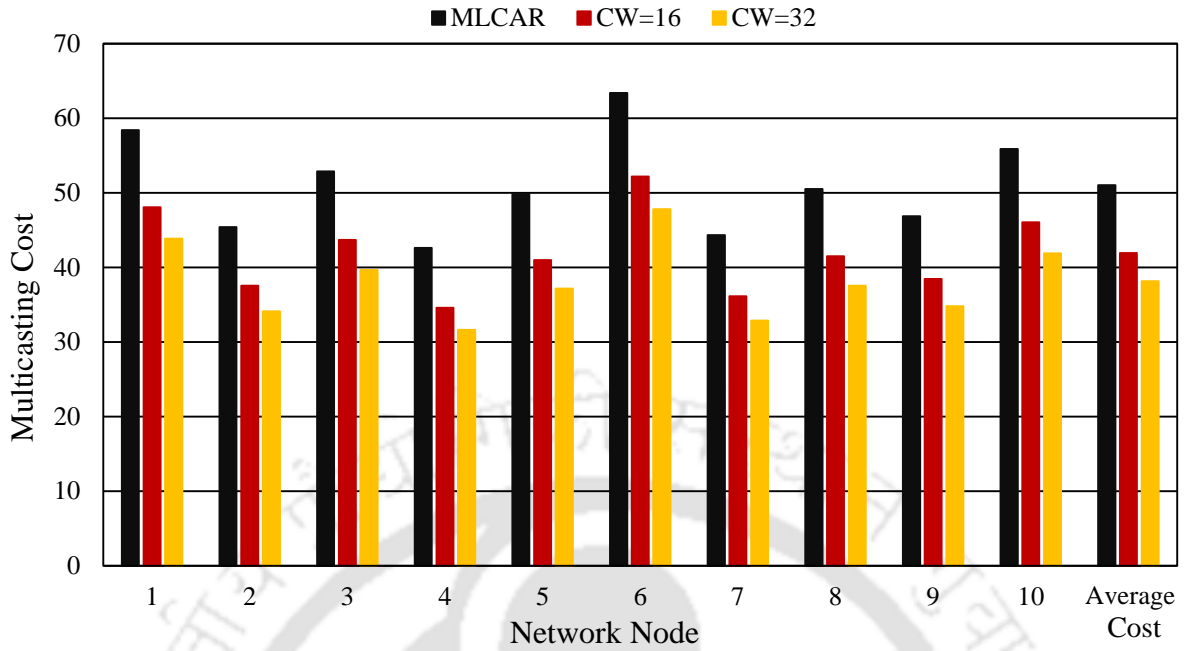


Figure 5.4: Multicasting cost of 10 random networks along with their average cost

#### 5.4.2 Simulation Results of NCRMM

An event-driven simulator is designed in the C++ programming language to determine multicasting costs for MLCAR, NC-MLCAR, and NCRMM schemes. For more effective simulation results, the 10 random topologies with 30 static nodes are generated. For each topology, four source nodes and the corresponding destination sets of size 4 are randomly selected. The successful packet delivery probabilities  $p_s$  of the network links are randomly generated in the range  $\{0.10, 0.90\}$ ; the bidirectional links are assumed to have the same  $p_s$  values. A batch size (BS) of one implies packet wise transmission similar to that in case of MLCAR scheme. The batch size in NC-MLCAR is 32. In correlated link scenario, the probability of correlation between network links is based on  $p_s$ . Let  $\xi = \min\{p_{n,1}, \dots, p_{n,H}\}$  for any network node  $n$ , where  $p_{n,h}$ ,  $h = \{1, \dots, H\}$  is  $p_s$  for the link between  $n$  and  $h$ , and  $H$  is the size of neighboring nodes of node  $n$ ,  $NS_n$ . The effect of link correlation on the multicasting cost is presented for low link correlation,  $p_c = 10\%$  of  $\xi$ , and high link

correlation,  $p_c = 90\%$  of  $\xi$ , where  $p_c$  is the probability of correlation for all the possible combinations of links from node  $n$  to the corresponding nodes of  $NS_n$ .

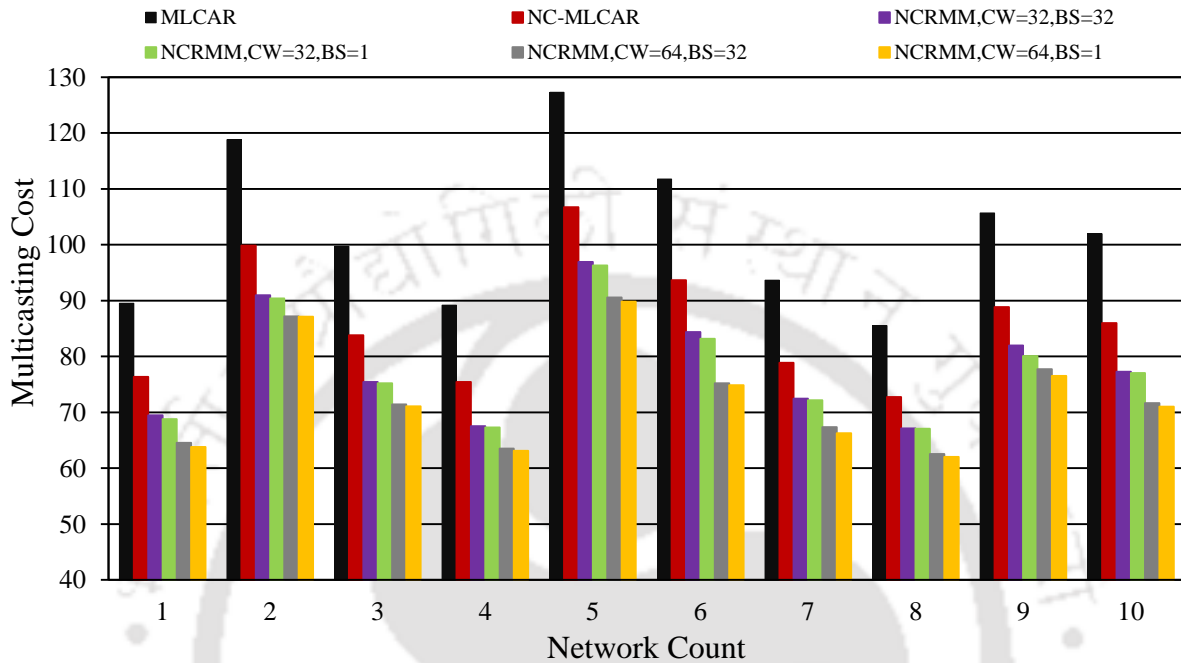


Figure 5.5: Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in independent link scenario

The ETXs needed to reliably send packets from all four source nodes to the respective destination sets is defined as the multicasting cost. The multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for independent links, low link correlation, and high link correlation scenarios are shown in Figs. 5.5, 5.6, and 5.7, respectively for 10 random topologies with 30 static nodes. Further, the average multicasting costs of MLCAR, NC-MLCAR, and NCRMM schemes are plotted for the three cases in Fig. 5.8. The average multicasting cost of the NCRMM scheme is less than that of the MLCAR and NC-MLCAR protocols. The average multicasting cost of MLCAR scheme reduces with an increase in the network links correlation. When the correlation between the links is high, the packets

received at nodes are almost identical. Further, the average multicasting cost of NC-MLCAR and NCRMM schemes is almost constant on varying link correlation.

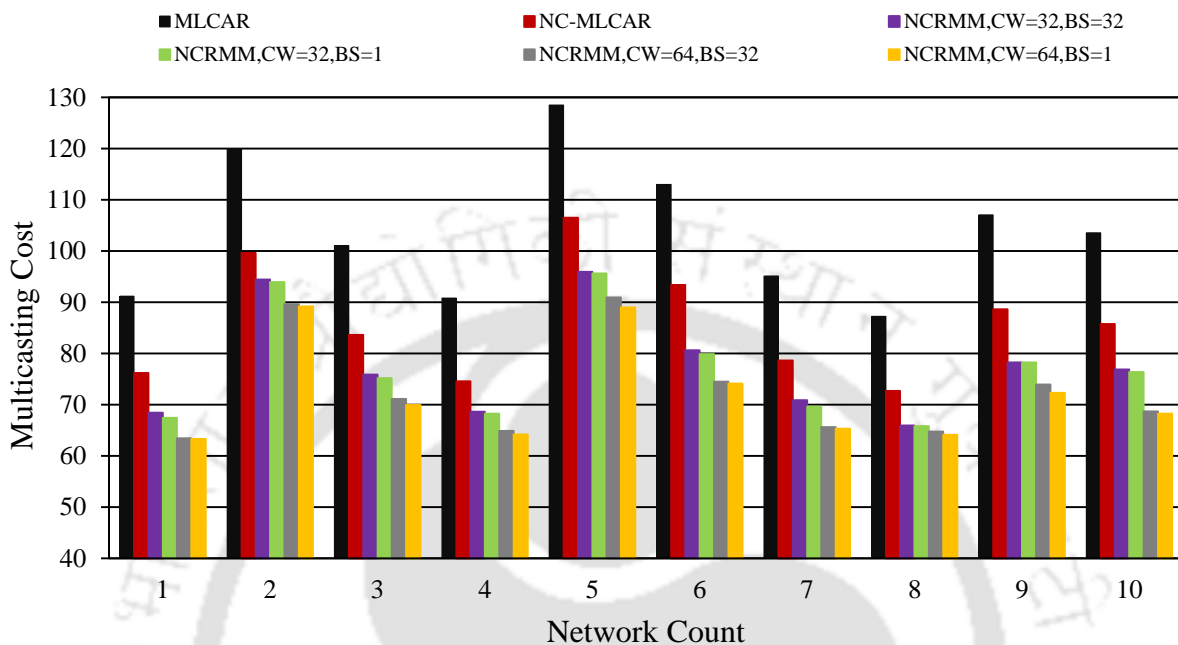


Figure 5.6: Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in low link correlation scenario

The increase in CW size gives more opportunity to combine packets and, therefore, reduce the overall transmissions in a network at the cost of some packets delay. As seen in Fig. 5.8, the performance improvement of NCRMM over MLCAR when  $CW = 64$  and Batch Size (BS) = 1 is around 29.13% for the network with independent links, 30.65% for the network with low link correlation, and 19.35% for the network with high link correlation. When  $CW = 64$ ,  $BS = 32$ , the performance improvement is around 28.50% for the network with independent links, 30% for the network with low link correlation, and 18.54% for the network with high link correlation. Thus, the proposed NCRMM scheme shows a significant advantage over the existing MLCAR scheme in both independent and correlated link networks.

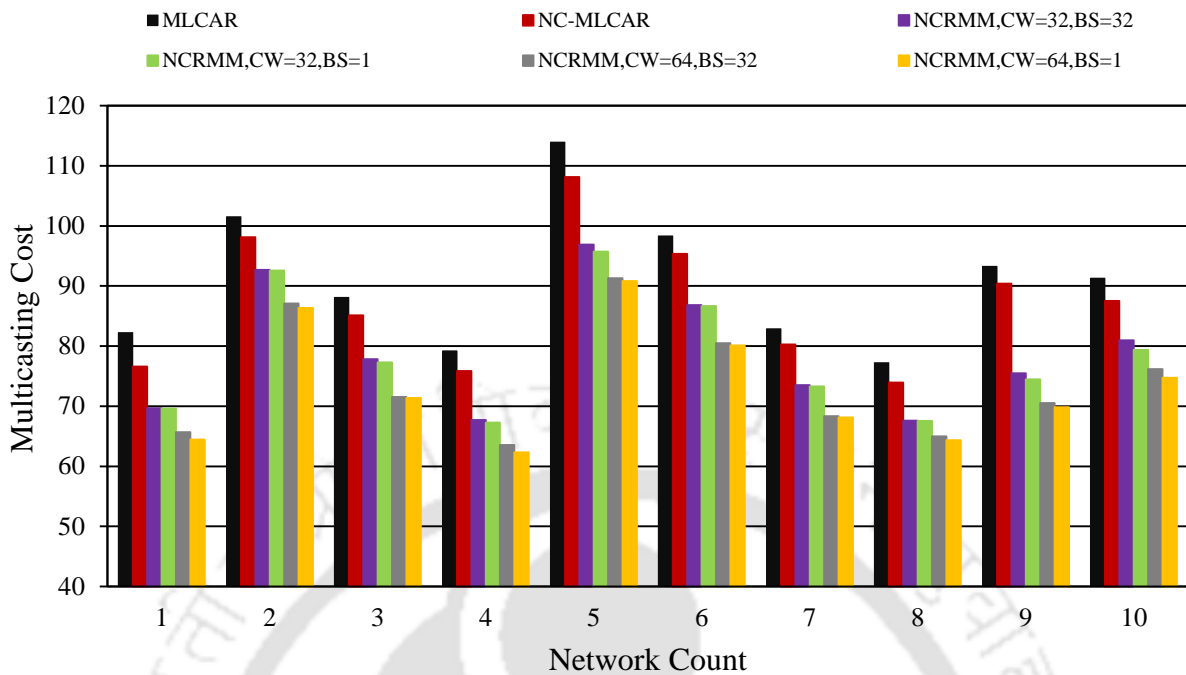


Figure 5.7: Multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes for 10 random topologies with 30 static nodes in high link correlation scenario

Further, the performance improvement of NCRMM over NC-MLCAR when  $CW = 64$ ,  $BS = 1$  is around 15.86% for the network with independent links, 16.30% for the network with low link correlation, 15.94% for the network with high link correlation. Similarly, when  $CW = 64$ ,  $BS = 32$ , the performance improvement is around 15.16% for the network with independent links, 15.40% for the network with low link correlation, 15.10% for the network with high link correlation. This implies that the proposed scheme, which supports simultaneous transmission from multiple sources, performs better than the NC-MLCAR scheme.

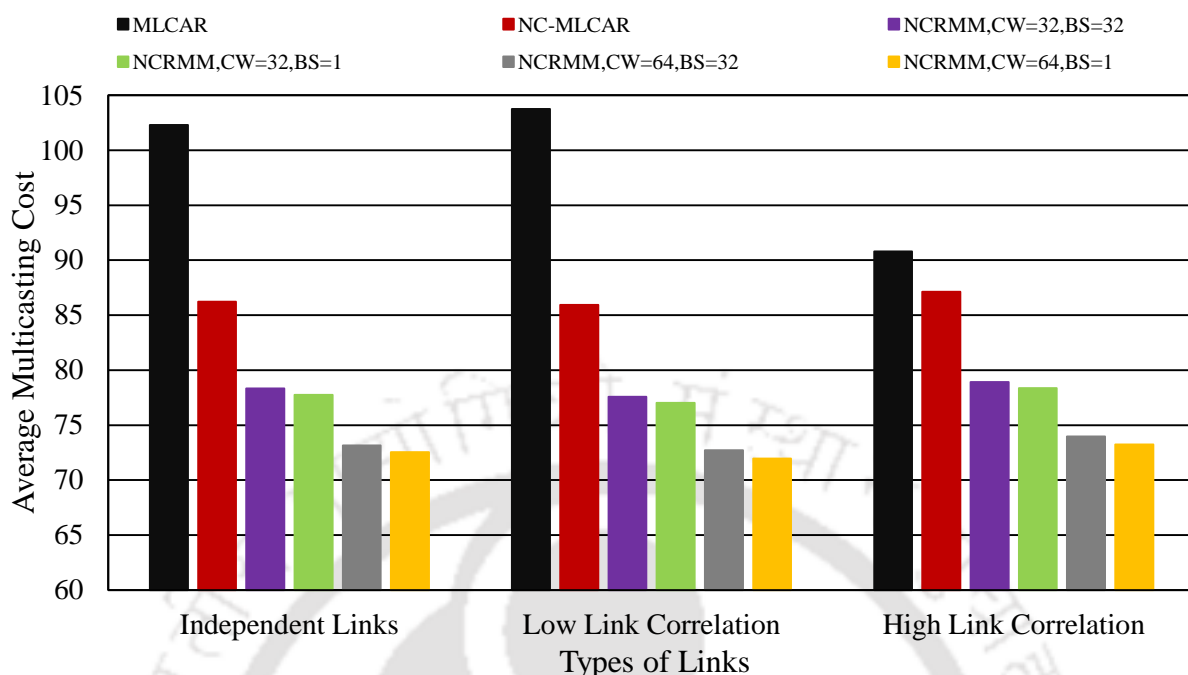


Figure 5.8: Average multicasting cost of MLCAR, NC-MLCAR, and NCRMM schemes

### 5.4.3 Complexity Analysis

The complexity of the NCRMM scheme can be expressed in terms of complexity incurred due to multicast cost calculation and packet transmission in a network. A Bellman-Ford type algorithm is used to calculate multicast route cost from network node  $n$  reaching subsets of  $DS_n^{(q)}$  for the  $q^{\text{th}}$  session through all possible combinations of neighboring nodes  $NS_n$ . Thus, the complexity is given by  $O(2^{(|NS_n|+|DS_n^{(q)}|)})$ . Suppose, a network node  $n$  reaches subsets of  $DS_n^{(q)}$  for  $m_c$  multicasting sessions where  $m_c \leq m_s$ , then the overall complexity is computed as  $O(\sum_{q=1}^{m_c} 2^{(|NS_n|+|DS_n^{(q)}|)})$ . Further, using Algorithms 1 – 4, efficient packet transmission is performed from the source nodes to the corresponding destination sets. In Algorithm 1, subsets of  $FS_n^{(q)}$  reaches every node of  $DS_n^{(q)}$  with complexity  $O(|FS_n^{(q)}|/|DS_n^{(q)}|)$  for the  $q^{\text{th}}$  multicasting session. The complexity incurred in storing subsets of  $FS_n^{(q)}$  with

$N^{(q)}$  innovative packets in  $ENE^{(q)}$  for the  $q^{\text{th}}$  multicasting session is  $O(|M_n^{(q)}|^2)$ , where  $M_n^{(q)}$  is size of the destination set belonging to  $q^{\text{th}}$  session. If the CW at network node  $n$  has packets of  $m_p$  sessions, where  $m_p \leq m_s$ , the total complexity of Algorithm 1 is bounded by

$$O\left(\sum_{q=1}^{m_p} (|FS_n^{(q)}| |DS_n^{(q)}| + |M_n^{(q)}|^2)\right).$$

When node  $n$  is the source node for the single multicasting session then  $m_p = 1$ . Using Algorithm 2, node  $n$  communicates network-coded packets that are innovative for the maximum number of nodes present in  $EFSN$ . Network-coded packets are formed using packet reception status of the  $N_{cw}$  innovative packets transmitted to nodes of  $EFSN$ , this incurs complexity of  $O(N_{cw} \| EFSN |)$  where  $N_{cw}$  is the number of packets present in the CW. In the case of source nodes,  $N_{cw}$  is the number of packets in a batch. In

Algorithm 3, a node searches for every possible combination of  $FSU^{(q)}$  reaching subsets of  $DS_n^{(q)}$  such that the overall multicasting cost is minimum. Since  $FS_n^{(q)}$  is the largest  $FSU^{(q)}$  therefore the complexity of Algorithm 3 for the  $m_p$  multicasting sessions in CW is bounded

$$\text{by } O\left(\sum_{q=1}^{m_p} 2^{(|FS_n^{(q)}| + |DS_n^{(q)}|)}\right).$$

For a multicasting session, Algorithm 4 assigns nodes in  $F_n^{(q)}$  as new  $n$  and the respective destination set present in  $D_n^{(q)}$  as  $DS_n^{(q)}$ . The corresponding innovative packets present in  $IP_n^{(q)}$  are assigned to  $CW_n$ . This is done for  $m_p$  multicasting

sessions resulting in the total complexity of  $O\left(\sum_{q=1}^{m_p} |FS_n^{(q)}| (|F_n^{(q)}| + |D_n^{(q)}|)\right)$ . The complexity

analysis of the proposed NCRMM scheme is tabulated along with that for MLCAR and NC-MLCAR schemes. It is observed that the complexity grows with increasing number of multicasting sessions.

Table 5.1: Comparison of complexity analysis of MLCAR, NC-MLCAR, and NCRMM

	Algorithm complexity		
	MLCAR	NC-MLCAR	NCRMM
Route cost calculation	$O(2^{( NS_n + DS_n^{(1)} )})$	$O(2^{( NS_n + DS_n^{(1)} )})$	$O(\sum_{q=1}^{m_c} 2^{( NS_n + DS_n^{(q)} )})$
Packet transmission	$O(2^{( FS_n^{(1)} + DS_n^{(1)} )})$	$O(2^{( FS_n^{(1)} + DS_n^{(1)} )})$ $+O( FS_n^{(1)}   DS_n^{(1)}  +  M_n^{(1)} ^2)$ $+O( N_n^{(1)}   FS_n^{(1)} )$ $+O( FS_n^{(1)}  ( F_n^{(1)}  +  D_n^{(1)} ))$	$O(\sum_{q=1}^{m_p} 2^{( FS_n^{(q)} + DS_n^{(q)} )})$ $+O(\sum_{q=1}^{m_p} ( FS_n^{(q)}   DS_n^{(q)}  +  M_n^{(q)} ^2))$ $+O( N_{cw}    EFSN )$ $+O(\sum_{q=1}^{m_p}  FS_n^{(q)}  ( F_n^{(q)}  +  D_n^{(q)} ))$

## 5.5 Summary

In this chapter, an efficient multicasting strategy is introduced for static multi-source multi-hop wireless networks with independent and correlated links. The neighboring nodes participating in packet transmission are selected using the MLCAR protocol [5]. Each source node multicast packets to the respective forwarding nodes using intra-session NC. On multicasting, the packets received at the forwarding nodes can be from the same or different sessions which are stored in the CW of the node. These packets are combined using intra-session and inter-session NC. The status of packets received from the neighboring nodes is communicated using ACKs/NACKs. This ensures that the innovative packets of all the multicasting sessions are reliably transmitted to the selected neighboring nodes, which then communicate the packets to the desired destination sets. A network-coded packet is formed and transmitted such that it is innovative for the maximum number of selected neighboring

nodes. When the desired forwarding nodes successfully receive the innovative packets of a session, these packets are judiciously distributed among the nodes so that the MLCAR cost to reach the corresponding destination set is minimum. Extensive simulations are performed to compare the performance of the proposed NCRMM scheme with that of the corresponding MLCAR and NC-MLCAR protocols. A significant reduction is found in the overall transmission cost of the NCRMM scheme for both correlated and independent link networks.







# Chapter 6

## Conclusions and Future Scopes

This chapter concludes the thesis, which mainly focuses on reliable multicasting over multi-hop wireless networks. The proposed protocols consider networks with independent and correlated links. NC assisted transmission protocols are presented, which improve multicasting efficiency by reducing the expected number of overall transmissions in the network. Some future directions that can be investigated further are also summarized.

### 6.1 Conclusions

The MLCAR scheme uses the Bellman-Ford approach to calculate the multicasting cost from the source to all possible subsets of the destination set and packet transmission is performed over the least-cost multicasting route obtained using the MLCAR protocol. Most authors have assumed that all the wireless network links are independent, but in a practical scenario, there may be finite correlations between different links. The links of a network are considered highly correlated if the receiving nodes have higher chances of successfully receiving the packet in the same transmission. When links have a low correlation, the probability of the packet traversing the links correctly in the same transmission would be low. In this thesis, a correlated multicast network is considered where forwarder set selection is made using the MLCAR approach. A generalized expression to calculate multicasting cost and determine the associated forwarder set for a correlated link environment is presented. The multicasting path obtained is used to transmit the packet efficiently from the source node to the designated destination set. The simulation results show that the multicasting cost and its corresponding routes are significantly different when compared to that for a network with independent links.

NC is a useful technique that takes into account the information content of each transmission and uses that knowledge to improve network throughput. The proposed NC-MLCAR scheme is applicable to both independent and correlated link networks. The packet reception status is tracked using ACK/NACK, which ensures that all the innovative packets are forwarded to the successive nodes and ultimately reach the destination set. The combined packets are innovative for the maximum number of selected receiving nodes and distribute packets among the forwarding nodes judiciously so that the cost to reach the destination set is minimum. Simulation results show the NC-MLCAR scheme's advantage over the existing NC assisted multicasting approaches such as MORE and Pacifier. The results are also compared with the corresponding MLCAR approach. A significant reduction in multicasting cost is observed for the proposed NC-MLCAR scheme. In a practical scenario, links may not be independent, so simulation results considering link correlations are also presented for the different schemes.

Multi-source multicasting is an emerging topic due to the massive demand for applications where clients request services like billing records, medical images, and new software updates. The MLCAR protocol gives the least multicasting cost and the corresponding route associated with it. The MLCAR protocol is used to determine the multicasting cost and the forwarder set for each of the sessions. The packet transmission employs NC, using coding coefficients 0 and 1. Source node does batch-wise communication and multicasts packets to the respective forwarding nodes using intra-session NC. Each forwarding node has a CW where packets of different sessions are stored, and therefore, it allows performing intra-session and inter-session NC. The packet reception status at the neighboring nodes are communicated using ACK/NACK, which ensures that the innovative packets of all the multicasting sessions are transmitted to the forwarding nodes, which then sends them to the

associated destination sets. A network coded packet is formed and transmitted such that it is innovative for the maximum number of selected neighboring nodes. The selected forwarding nodes successfully receive the innovative packets of a session and the packets are distributed judiciously among them so that the MLCAR cost of reaching the corresponding destination set is the least. An event-driven simulator is designed in C++ to verify the performance gain over the corresponding MLCAR and NC-MLCAR protocols. A significant reduction in the overall transmission cost of the proposed scheme is found for both independent and correlated link networks.

## **6.2 Future Scopes**

This thesis presented NC assisted routing protocols for multicasting in a multi-hop wireless network where the network nodes are assumed to be static. Extending this, multicasting routing protocols for a dynamically changing network may also be designed and implemented. In this thesis, the forwarder set selection is made using the MLCAR protocol as it provides the least multicasting cost from source to the destination set, even though as the destination set size increases, the complexity for cost calculation and forwarder set selection would increase. Therefore, heuristic approaches for multicasting cost calculation and forwarder set selection may be looked into in order to reduce the complexity.

The focus of this thesis is to perform reliable multicasting where strict packet reliability is needed at all the intended receivers, such as in software updates, medical images, and downloading a new security patch. Unreliable multicasting routing protocols in multi-hop wireless networks may also be designed and investigated for applications like live video streaming, which requires packet delivery within a predefined threshold time. The thesis has individually considered reliable multi-source multicasting routing protocols for multi-hop wireless networks. Multi-source multicasting routing protocol for unreliable hard-deadline

applications may be further investigated. The presence of unwanted malicious node may affect the packet transmissions in the network. Therefore, implementation of a secure single-source and multi-source multicasting schemes can be investigated in the future.







# References

- [1] S. Biswas and R. Morris, "ExOR: Opportunistic multi-hop routing for wireless networks," in *Proceedings of the ACM Annual Conference of the Special Interest Group on Data Communication*, Philadelphia, Pennsylvania, USA, August 21-26 2005, pp. 133–144.
- [2] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Valuable detours: Least-cost anypath routing," *IEEE/ACM Transactions on Networking*, vol. 19, no. 2, pp. 333-346, April 2011.
- [3] S. Wang, A. Basalamah, S. M. Kim, S. Guo, Y. Tobe, and T. He, "Link correlation-aware opportunistic routing in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 1, pp. 47-56, January 2015.
- [4] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *Proceedings of the ACM Special Interest Group on Data Communication*, Kyoto, Japan, August 27-31 2007, pp. 169-180.
- [5] S. P. Mazumdar, S. K. Bose, and W. D. Zhong, "Multicast least cost anypath routing: Route cost calculations and forwarding," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1652-1655, October 2012.
- [6] G. Feng, "GRASP-BSMA: A fast algorithm for delay constrained multicast routing," in *Proceedings of the IEEE International Conference on Communications*, Ottawa, Canada, June 10-15 2012, pp. 2833-2837.
- [7] D. Koutsonikolas, Y. C. Hu, and C. C. Wang, "Pacifier: High-throughput, reliable multicast without crying babies in wireless mesh networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1375-1388, October 2012.
- [8] A. Zubow, M. Kurth, and J. P. Redlich, "Considerations on forwarder selection for opportunistic protocols in wireless networks," in *Proceedings of the Fourteenth European Wireless Conference*, Prague, June 22-25 2008, pp. 1-7.
- [9] K. Srinivasan, M. Jain, J. I. Choi, T. Azim, E. S. Kim, P. Levis, and B. Krishnamachari, "The  $\kappa$  factor: Inferring protocol performance using inter-link reception correlation," in *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking*, New York, USA, September 20-24 2010, pp. 317-328.
- [10] Z. Zhao et al., "Exploiting link correlation for core-based dissemination in wireless sensor networks," in *Proceedings of the Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking*, Singapore, June 30-July 3 2014, pp. 372-380.

- [11] X. Shen et al., "OppCode: Correlated opportunistic coding for energy-efficient flooding in wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 6, pp. 1631-1642, December 2015.
- [12] T. N. Weerasinghe, A. Kanyeshuli, and F. Y. Li, "Evaluation and improvement of collective flooding in WSNs with various link correlations," in *Proceedings of the Twenty Sixth Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, Hong Kong, August 30-September 2 2015, pp. 1244-1249.
- [13] G. Huang, B. Zhang, and Z. Yao, "Data correlation aware opportunistic routing protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications*, Paris, May 21-25 2017, pp. 1-6.
- [14] S. M. Kim, S. Wang, and T. He, "Exploiting causes and effects of wireless link correlation for better performance," in *Proceedings of the IEEE Conference on Computer Communications*, Kowloon, April 26-May 1 2015, pp. 379-387.
- [15] Z. Zhao, W. Dong, G. Guan, J. Bu, T. Gu, and C. Chen, "Modeling link correlation in low-power wireless networks," in *Proceedings of the IEEE Conference on Computer Communications*, Kowloon, April 26-May 1 2015, pp. 990-998.
- [16] S. Wang, A. Basalamah, S. M. Kim, G. Tan, Y. Liu, and T. He, "A unified metric for correlated diversity in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 9, pp. 6215-6227, September 2016.
- [17] S. M. Kim, S. Wang, and T. He, "Exploiting spatiotemporal correlation for wireless networks under interference," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3132-3145, October 2017.
- [18] A. Kamari and M. Bag-Mohammadi, "An optimized link correlation model for opportunistic routing," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2543-2546, December 2018.
- [19] C. Caol, W. Gong, W. Dong, J. Yu, C. Chen, and J. Liu, "Network measurement in multi-hop wireless networks with lossy and correlated links," in *Proceedings of the IEEE Conference on Computer Communications*, Honolulu, April 16-19 2018, pp. 1340-1348.
- [20] H. Shen, J. Xu, T. Wang, and G. Bai, "Detecting link correlation spoofing attack: A beacon-trap approach," in *Proceedings of the IEEE International Conference on Communications*, Shanghai, China, May 20-24 2019, pp. 1-6.
- [21] V. Chheda, S. K. Bose, and W. Zhong, "MC-OR: Opportunistic routing based multicast protocol for wireless networks," in *Proceedings of the 8th International Conference on Information, Communications & Signal Processing*, Singapore, December 13-16 2011, pp. 1-6.

- [22] D. Li, Y. Li, J. Wu, S. Su, and J. Yu, "ESM: Efficient and scalable data center multicast routing," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 944-955, June 2012.
- [23] W. Jia and L. Wang, "A unified unicast and multicast routing and forwarding algorithm for software-defined data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 12, pp. 2646-2657, December 2013.
- [24] W. Jia, "A scalable multicast source routing architecture for data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 1, pp. 116-123, January 2014.
- [25] Z. Guo, J. Duan, and Y. Yang, "On-line multicast scheduling with bounded congestion in fat-tree data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 1, pp. 102-115, January 2014.
- [26] K. Kim, S. Mehrotra, and N. Venkatasubramanian, "Efficient and reliable application layer multicast for flash dissemination," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 10, pp. 2571-2582, October 2014.
- [27] X. Zhao, J. Guo, C. T. Chou, A. Misra, and S. K. Jha, "High-throughput reliable multicast in multi-hop wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 728-741, April 2015.
- [28] D. Papsdorf, Y. C. B. Silva, and A. Klein, "Multicast interference alignment in a multi-group multi-way relaying network," in *Proceedings of the IEEE International Conference on Communications*, Paris, May 21-25 2017, pp. 1-6.
- [29] E. Chen and M. Tao, "A fast algorithm for multi-group multicast beamforming in large-scale wireless systems," in *Proceedings of the IEEE International Conference on Communications*, Paris, May 21-25 2017, pp. 1-6.
- [30] L. Yang, D. Wu, Y. Zhang, C. Yue, Y. Wu, and X. Shi, "Dynamic link selection for interference mitigation in D2D multicast content sharing networks," *IEEE Communications Letters*, vol. 24, no. 1, pp. 217-221, October 2020.
- [31] T. V. Santana, R. Combes, and M. Kobayashi, "Performance analysis of device-to-device aided multicasting in general network topologies," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 137-149, January 2020.
- [32] J. Zhang, P. Fan, and K. B. Letaief, "Network coding for efficient multicast routing in wireless ad-hoc networks," *IEEE Transactions on Communications*, vol. 56, no. 4, pp. 598-607, April 2008.

- [33] P. Fan, C. Zhi, C. Wei, and K. B. Letaief, "Reliable relay assisted wireless multicast using network coding," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 749-762, June 2009.
- [34] D. Traskov, M. Heindlmaier, M. Medard, and R. Koetter, "Scheduling for network-coded multicast," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1479-1488, October 2012.
- [35] T. Wiese, M. Riemensberger, and W. Utschick, "Scheduling for network-coded multicast with interference," *IEEE Transactions on Signal Processing*, vol. 64, no. 9, pp. 2245-2254, May 2016.
- [36] B. Shrader and A. Ephremides, "Queueing delay analysis for multicast with random linear coding," *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 421-429, January 2012.
- [37] M. X. Cheng, Q. Ye, X. Cheng, and R. F. Erbacher, "Network coding and coding-aware scheduling for multicast in wireless networks," in *Proceedings of the IEEE International Conference on Communications*, London, June 8-12 2015, pp. 5703-5708.
- [38] C. Chen, S. J. Baek, and G. de Veciana, "Opportunistic scheduling of randomly coded multicast transmissions at half-duplex relay stations," *IEEE Transactions on Information Theory*, vol. 62, no. 10, pp. 5538-5555, October 2016.
- [39] C. Chen and S. J. Baek, "Multicast scheduling for relay-based heterogeneous networks using rateless codes," *IEEE Transactions on Mobile Computing*, vol. 16, no. 11, pp. 3142-3155, November 2017.
- [40] K. C. Lin and D. Yang, "Multicast with intra-flow network coding in multi-rate multichannel wireless mesh networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 8, pp. 3913-3927, October 2013.
- [41] F. Chiti, R. Fantacci, F. Schoen, and A. Tassi, "Optimized random network coding for reliable multicast communications," *IEEE Communications Letters*, vol. 17, no. 8, pp. 1624-1627, August 2013.
- [42] E. Tsimballo, A. Tassi, and R. J. Piechocki, "Reliability of multicast under random linear network coding," *IEEE Transactions on Communications*, vol. 66, no. 6, pp. 2547-2559, June 2018.
- [43] G. Yue, K. Yang, S. Zhao, and H. V. Poor, "Design of network coding for wireless broadcast and multicast with optimal decoders," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 6944-6957, October 2018.
- [44] Y. Chen, L. Wang, K. Wang, and W. Ho, "Topology-aware network coding for wireless multicast," *IEEE Systems Journal*, vol. 12, no. 4, pp. 3683-3692, December 2018.

- [45] A. Douik, M. S. Al-Abiad, and M. J. Hossain, "An improved weight design for unwanted packets in multicast instantly decodable network coding," *IEEE Communications Letters*, vol. 23, no. 11, pp. 2122-2125, November 2019.
- [46] D. Jiang, Z. Xu, W. Li, C. Yao, Z. Lv, and T. Li, "An energy-efficient multicast algorithm with maximum network throughput in multi-hop wireless networks," *Journal of Communications and Networks*, vol. 18, no. 5, pp. 713-724, October 2016.
- [47] C. Hu, H. Wu, and G. Chen, "Bandwidth-satisfied multicast trees in MANETs," *IEEE Transactions on Mobile Computing*, vol. 7, no. 6, pp. 712-723, June 2008.
- [48] S. Bitam and A. Mellouk, "MQBM: An autonomic QoS multicast routing protocol for mobile ad hoc networks," in *Proceedings of the IEEE International Conference on Communications*, Ottawa, ON, June 10-15 2012, pp. 5488-5492.
- [49] G. Feng, "Delay constrained multicast routing: What can we learn from an exact approach?," in *Proceedings of the IEEE Global Communications Conference*, Anaheim, CA, December 3-7 2012, pp. 2809-2814.
- [50] G. Feng, "GRASP-BSMA: A fast algorithm for delay constrained multicast routing," in *Proceedings of the IEEE International Conference on Communications*, Ottawa, ON, June 10-15 2012, pp. 2833-2837.
- [51] T. Lu and J. Zhu, "Genetic algorithm for energy-efficient QoS multicast routing," *IEEE Communications Letters*, vol. 17, no. 1, pp. 31-34, January 2013.
- [52] S. Wang and W. Liao, "Cooperative multicasting for wireless scalable video transmissions," *IEEE Transactions on Communications*, vol. 61, no. 9, pp. 3980-3989, September 2013.
- [53] M. S. Rahman and A. B. Wagner, "A downlink scheduler for multicasting wireless video-on-demand," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 2921-2938, December 2016.
- [54] Z. Chen, P. Ho, and L. Peng, "Distortion minimization for relay-assisted wireless multicast," *Journal of Communications and Networks*, vol. 20, no. 1, pp. 1-8, February 2018.
- [55] T. Li, H. Zhang, X. Zhou, and D. Yuan, "NOMA enabled layered video multicast in wireless-powered relay systems," *IEEE Communications Letters*, vol. 23, no. 11, pp. 2118-2121, November 2019.
- [56] M. Li and Y. Wu, "Performance analysis of adaptive multicast streaming services in wireless cellular networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 11, pp. 2616-2630, November 2019.

- [57] S. Wang, C. Gong, X. Wang, and M. Liang, "An efficient retransmission strategy for wireless scalable video multicast," *IEEE Wireless Communications Letters*, vol. 1, no. 6, pp. 581-584, December 2012.
- [58] B. Li, H. Li, and R. Zhang, "Adaptive random network coding for multicasting hard deadline constrained prioritized data," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 8739-8744, October 2016.
- [59] H. Li, B. Li, T. T. Tran, and D. C. Sicker, "Transmission schemes for multicasting hard deadline constrained prioritized data in wireless multimedia streaming," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 1631-1641, March 2016.
- [60] A. Tassi, I. Chatzigeorgiou, and D. E. Lucani, "Analysis and optimization of sparse random linear network coding for reliable multicast services," *IEEE Transactions on Communications*, vol. 64, no. 1, pp. 285-299, January 2016.
- [61] B. Li, X. Li, R. Zhang, W. Tang, and S. Li, "Joint power allocation and adaptive random network coding in wireless multicast networks," *IEEE Transactions on Communications*, vol. 66, no. 4, pp. 1520-1533, April 2018.
- [62] S. Shen, "Efficient SVC multicast streaming for video conferencing with SDN control," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 403-416, June 2019.
- [63] W. Yeow, A. T. Hoang, and C. Tham, "Minimizing delay for multicast-streaming in wireless networks with network coding," in *Proceedings of the IEEE International Conference on Computer Communications*, Rio de Janeiro, April 19-25 2009, pp. 190-198.
- [64] K. Rajawat and G. B. Giannakis, "Joint scheduling and network coding for multicast in delay-constrained wireless networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6186-6196, December 2011.
- [65] H. Lu and W. Liao, "Cooperative multicasting in network-coding enabled multi-rate wireless relay networks," in *Proceedings of the IEEE International Conference on Computer Communications*, Orlando, FL, March 25-30 2012, pp. 334-342.
- [66] S. Lakshminarayana and A. Eryilmaz, "Multi-rate multicasting with intra-layer network coding," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1256-1269, August 2013.
- [67] H. L. Nguyen and U. T. Nguyen, "Performance evaluation of network-coded multicast in multi-channel multi-radio wireless mesh networks," in *Proceedings of the IEEE International Conference on Communications*, Ottawa, ON, June 10-15 2012, pp. 664-670.

- [68] Y. Chen, E. H. Wu, and G. Chen, "Bandwidth-satisfied multicast by multiple trees and network coding in lossy MANETs," *IEEE Systems Journal*, vol. 11, no. 2, pp. 1116-1127, June 2017.
- [69] Y. Chen, E. H. Wu, C. Lin, and G. Chen, "Bandwidth-satisfied and coding-aware multicast protocol in MANETs," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1778-1790, August 2018.
- [70] K. Choumas, I. Syrigos, T. Korakis, and L. Tassiulas, "Video-aware multicast opportunistic routing over 802.11 two-hop mesh networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 8372-8384, September 2017.
- [71] B. Li, H. Li, X. Li, H. Jiang, W. Tang, and S. Li, "Hybrid multicast and device-to-device communications based on adaptive random network coding," *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 2071-2083, March 2019.
- [72] S. M. Banik and Z. C. Aardahl, "M3DDVC: Multi-source multicasting using multi-cores with delay and delay variation constraints on overlay networks," in *Proceedings of the Eighth International Conference on Information Technology: New Generations*, Las Vegas, NV, April 11-13 2011, pp. 661-666.
- [73] Yuh-Rong Chen, S. Radhakrishnan, S. K. Dhall, and S. Karabuk, "On multi-stream multi-source multicast routing," in *Proceedings of the IEEE Global Communications Conference*, Anaheim, CA, December 3-7 2012, pp. 1926-1931.
- [74] A. Argyriou, "Link scheduling for multiple multicast sessions in distributed wireless networks," *IEEE Wireless Communications Letters*, vol. 2, no. 3, pp. 343-346, June 2013.
- [75] W. Tu, "Efficient multi-flow multicasting in wireless multi-hop networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Shanghai, April 7-10 2013, pp. 2232-2237.
- [76] J. J. Garcia-Luna-Aceves, "Efficient multi-source multicasting in Information Centric Networks," in *Proceedings of the 12th Annual IEEE Consumer Communications and Networking Conference*, Las Vegas, NV, January 9-12 2015, pp. 245-249.
- [77] W. Tu, C. J. Sreenan, S. Jha, and Q. Zhang, "Multi-source video multicast in internet-connected wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 12, pp. 3431-3444, December 2017.
- [78] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497-510, June 2008.

- [79] R. S. Youail, C. Wenqing, and T. Shaoguo, "Cost minimization for multi-source multi-sinks network coding," in *Proceedings of the 9<sup>th</sup> International Conference for Young Computer Scientists*, Hunan, November 18-21 2008, pp. 253-258.
- [80] Z. Ding, Z. Ma, and K. K. Leung, "Impact of network coding on system delay for multi-source multi-destination scenarios," *IEEE Transactions on Vehicular Technology*, pp. 831-841, vol. 59, no. 2, February 2010.
- [81] G. S. Paschos, C. Fragiadakis, L. Georgiadis, and L. Tassiulas, "Wireless network coding with partial overhearing information," in *Proceedings of the IEEE International Conference on Computer Communications*, Turin, April 14-19 2013, pp. 2337-2345.
- [82] A. Khreishah, J. Wu, P. Ostovari, and I. M. Khalil, "Flow based XOR network coding for lossy wireless networks," in *Proceedings of the IEEE Global Telecommunications Conference*, Kathmandu, December 5-9 2011, pp. 1-5.
- [83] H. Seferoglu, A. Markopoulou, and K. K. Ramakrishnan, "I2NC: Intra- and inter-session network coding for unicast flows in wireless networks," in *Proceedings of the IEEE International Conference on Computer Communications*, Shanghai, April 10-15 2011, pp. 1035-1043.
- [84] K. Ou, Y. Xu, and S. Fu, "Rate selection for wireless networks with intra and inter-session network coding," in *Proceedings of the IEEE Global Communications Conference*, Anaheim, CA, December 3-7 2012, pp. 4707-4712.
- [85] A. Khreishah, I. Khalil, and J. Wu, "Low complexity and provably efficient algorithm for joint inter and intra-session network coding in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 10, pp. 2015-2024, October 2013.
- [86] X. Li, C. Wang, and X. Lin, "Inter-session network coding schemes for 1-to-2 downlink access-point networks with sequential hard deadline constraints," *IEEE/ACM Transactions on Networking*, pp. 624-638, vol. 25, no. 1, February 2017.
- [87] T. Kunz, K. Mahmood, and L. Li, "Broadcasting in multi-hop wireless networks: The case for multi-source network coding," in *Proceedings of the IEEE International Conference on Communications*, Ottawa, ON, June 10-15 2012, pp. 5157-5162.
- [88] X. Deng and Y. Yang, "An efficient MAC multicast protocol for reliable wireless communications with network coding," in *Proceedings of the IEEE Global Telecommunications Conference*, Kathmandu, December 5-9 2011, pp. 1-6.
- [89] B. Pu, L. Yang, W. Wang, and X. Xiao, "Linear network coding construction for multi-source multicast network," in *Proceedings of the First International Workshop on Education Technology and Computer Science*, Wuhan, Hubei, March 7-8 2009, pp. 114-118.

- [90] T. Ho and H. Viswanathan, "Dynamic algorithms for multicast with intra-session network coding," *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 797-815, February 2009.
- [91] S. Shetty and A. E. Kamal, "Hybrid network-erasure coding protection of multi-source, multi-sink multicast sessions in WSNs," in *Proceedings of the IEEE Global Communications Conference*, Anaheim, CA, December 3-7 2012, pp. 131-137.
- [92] D. Lee, K. Kim, and J. K. Rhee, "Optimal multicast control for simple network coding," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 6, pp. 2375-2386, June 2015.
- [93] M. Kwon and H. Park, "Network coding-based distributed network formation game for multi-source multicast networks," in *Proceedings of the IEEE International Conference on Communications*, Paris, May 21-25 2017, pp. 1-6.
- [94] R. Zhao, J. Wang, K. Lu, X. Chang, J. Jia, and S. Zhang, "Optimal transmission topology construction and secure linear network coding design for virtual-source multicast with integral link rates," *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3069-3083, November 2018.
- [95] A. Cohen, A. Cohen, M. Medard, and O. Gurewitz, "Secure multi-source multicast," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 708-723, January 2019.



# List of Publications

## Journals

1. Prateek Rathore, Kalpana Dhaka, and Sanjay K. Bose, “Network coding assisted multicasting in multi-hop wireless networks,” in *Computer Communications*, vol. 138, pp. 45-53, 2019.
2. Prateek Rathore, Kalpana Dhaka, and Sanjay K. Bose, “Network coding assisted reliable multi-source multicasting over a multi-hop wireless networks,” in *Computer Communications* (under review).

## Conferences

1. Prateek Rathore, Kalpana Dhaka, and Sanjay K. Bose, “Multicasting in wireless networks with correlated links,” in *Proceedings of the IEEE Region 10 Conference (TENCON)*, Singapore, November 22-25 2016, pp. 1798-1802.
2. Prateek Rathore, Kalpana Dhaka, and Sanjay K. Bose, “Network coding assisted reliable multicasting in multi-hop wireless networks with two sources,” in *Proceedings of the National Conference on Communications*, Kharagpur, India, February 21-23 2020, pp. 1-6.

