REDUCTION OF FORWARDING NODES TO REDUCE EFFECT OF BROADCAST STORM PROBLEM USING NETWORK CODING

Thesis submitted in fulfillment of the requirements for the Degree of

Doctor of Philosophy

By

Mayank Kumar Goyal

136207

Under the supervision of

Prof. Dr. S. P. Ghrera

&

Prof. Dr. JP Gupta

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY

WAKNAGHAT, SOLAN, H.P.

DECEMBER 2019

DECLARATION

I hereby declare that the work reported in the Ph.D. thesis entitled "**Reduction of Forwarding Nodes to Reduce Effect of Broadcast Storm Problem using Network Coding**" submitted at **Jaypee University of Information Technology, Waknaghat, Solan, H.P., India,** is an authentic record of my work carried out under the supervision of **Prof. Dr. S.P. Ghrera and Prof. Dr. JP Gupta.** I have not submitted this work elsewhere for any other degree or diploma. I am fully responsible for the content of my Ph.D Thesis.

Mayank Kumar Goyal

Department of Computer Science and Engineering.

Jaypee University of Information Technology, Waknaghat, Solan, H.P., India.

Date

SUPERVISOR'S CERTIFICATE

I hereby declare that the work reported in the Ph.D. thesis entitled "**REDUCTION OF FORWARDING NODES TO REDUCE EFFECT OF BROADCAST STORM PROBLEM USING NETWORK CODING**" submitted by **Mayank Kumar Goyal** at **Jaypee University of Information Technology, Waknaghat, Solan, H.P.**, **India** is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree or diploma.

Prof. Dr. S.P. Ghrera

Department of Computer Science and Engineering.

Jaypee University of Information Technology, Waknaghat, Solan, H.P., India.

Date

ACKNOWLEDGMENTS

Though only my name appears on the cover of this dissertation, a great many people have contributed to its production. I owe my gratitude to all those people who have made this dissertation possible and because of whom my graduate experience has been one that I will cherish forever. My deepest gratitude is to my advisor, Prof. Dr. S.P. Ghrera and Prof. Dr. JP Gupta. I have been amazingly fortunate to have an advisor who gave me the freedom to explore on my own and at the same time the guidance to recover when my steps faltered. His patience and support helped me overcome many crisis situations and finish this dissertation.

DPMC members, Dr. Pardeep Kumar, Dr. Rajinder Sindhu and Dr. Rajiv Kumar have been always there to listen and give advices. I am deeply grateful to all of them for the long discussions that helped me sort out the technical details of my work. I am also thankful to Dr. Vivek Sehgal, Dr. Pradeep Kumar Gupta, Dr. Punit Gupta, Dr. Amit Kumar, Dr. Rajni Mohana and Dr. Geetanjali for encouraging the use of proper timeline and for carefully reading and commenting on writing this manuscript. I am grateful to my colleagues for their encouragement and practical advices. I am also thankful to them for reading my reports, commenting on my views and helping me understanding and enrich my ideas. My sincere thanks to the undergraduate students and staff of the University, for their various forms of support during my study.

Many friends have helped me stay sane through these difficult years. Their support and care helped me overcome setbacks and stay focused on my graduate study. I greatly value their friendship and I deeply appreciate their belief in me. Most importantly, none of this would have been possible without the love and patience of my family, has been a constant source of love, concern, support and strength all these years.

Finally, I appreciate the financial and academic support of the Jaypee University of Information technology.

Mayank Kumar Goyal

Dept. of Computer Science and Engineering

Jaypee University of Information Technology

Waknaghat, Solan, H.P.

Dedication

I dedicate this dissertation to all my loved ones, those with me today and those who have passed on.

ABSTRACT

The Wireless mesh networks guarantee modest internet access, simple deployment, and extended range. In their present structure, however, these networks experience the ill effects of both limited throughput and high redundancy; hence they cannot fulfill the needs of applications such as file sharing and high definition video. Spurred by these problems, we explore an alternative design that addresses these challenges.

Message broadcasting is a basic function in wireless Ad-hoc network in which a node passes on a message to all neighbors thus causing redundant broadcast which is called as Broadcast storm Problem in which every node will be obligated to re-broadcast the data packet every time it gets the data packet thus causing redundancy. In this way, the number of forwarding nodes is utilized as the cost criterion for propagation. Instead of routers just storing and forwarding received packets, they mix (or code) packets' content before forwarding. We have developed few algorithms; each discloses a diverse advantage of our network coded design. Prior work on network coding focuses on multicast traffic. This thesis aims to bridge theory with practice by addressing the common issues faced in the integration of network coding in the current network stack .

The contributions of this dissertation are multifold. This dissertation presents an algorithm along with network coding concept that uses 1-Hop nodes to cover entire 2-Hop nodes utilizing 2-hop region information to decrease repetitive communicates. Simulation results of applying this algorithm demonstrate performance improvements. Now a day the scientists are acquainting the idea of Network coding to neighbour topology aware protocols that beats the excess number of broadcast by XOR of data packets. We have made an endeavor to seek out the network coding gain. We've shown simulation, implementation and breakdown of result in various circumstances.

Secondly, another proposed algorithm uses bit addresses to where every node is distinguished by bit value '1' in an address bit vector. Distinguishing packets and processing the network coding of packets can be effectively done utilizing address bit vectors. This algorithm also

acknowledges redundancy with total number of coded packets sent with respect to actual number of nodes present in the system. To carry out work in similar direction, Network Coding Algorithm also on integrating with the AODV protocol provides redundancy proficient system as compared to the old Network coding scheme since less number of average transmissions are required per node. The enhancement in the performance of the proposed AODV integrated Network coding scheme over the traditional Network coding scheme increases with increase in density. Hence a basic alteration of discovering a reduced sub-graph from the first original sub-graph utilizing AODV routing scheme can enhance the execution of Network Coding to an awesome degree .

At the same time, to achieve lesser redundancy in terms of achieving less number of forwarding nodes for retransmission uses the concept of degree of incoming traffic and outgoing traffic of a node along with link bandwidth when integrated with probabilistic based Network coding provides result to awesome degree. SET Partition theory based algorithm when integrated with Network Coding approach distinguishes and comprises nodes based on the fact of bundles of different network coded reduces the Number of Forwarding nodes for retransmission in contrast to actual number of nodes. Simultaneously, the design maintains desirable properties such as being distributed, implementable, and little complexity with the rest of the network stack.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

A mobile ad hoc network (MANET) is a set of versatile hubs that are powerfully and subjectively situated in such a way, that the interconnections between hubs are equipped for changing on a persistent premise. So as to encourage correspondence inside the system, a directing convention is utilized to find routes between hubs. The critical objective of such a specially appointed system directing convention is right & productive route foundation between a couple of hubs with the goal that messages might be conveyed in an opportune way. Route development ought to be finished with a minimum overhead & bandwidth utilization.

Wireless systems can be sorted into infrastructure and infrastructure-less dependent on the system arrangement and design. Versatile Ad-hoc Networks are foundation less systems with self-arranging property having a dynamic topology where hubs are allowed to move toward any path, because of this topological changes there are expanded odds of connection breaks in the system offering ascend to path failures and route disclosures. So as to find routes, hubs communicate course demand parcels to its neighbor, which thusly rebroadcast the bundles to their neighbors until they achieve the goal. This procedure start's Broadcast Storm Problem. Broadcasting assumes an unmistakable job for route revelations in MANET's since a large portion of the directing conventions utilize broadcasting strategies for route disclosures. Albeit numerous strategies have been proposed to address the issue of broadcast storms, they require extra occasional area reference points or don't tastefully decrease transmission repetition in high density conditions.

1.1 Advantages and disadvantages in wireless communication networks

One principle favorable position to a decentralized system is that they are ordinarily more powerful than centralized systems due to the multi-hop style in which data is transferred. For instance, in the cell arrange setting, a drop in inclusion happens if a base station quits working, anyway the possibility of a solitary purpose of disappointment in a MANET is diminished fundamentally since the data can take numerous routes. Since the MANET engineering develops with time it can possibly resolve issues, for example, disengagement from the system. Further points of interest of MANETS over systems with a fixed topology incorporate versatility and lower organization costs.

With these positives pursue some conspicuous disadvantages in system execution. Since network topology concludes interference & thus connectivity, mobility prototype of machines within network will bang on the performance of the network, probably resulting in increased delay and finally allocation of network resources. The Traffic Types in the Ad Hoc Networks

Peer to peer: correspondence between two hubs in a similar territory, that implies which are inside one jump.

Dynamic traffic: when the hubs are movie dynamically around & afterward the routers must be remade.

Remote to Remote: Communication between two hubs past a single hop, however keep up a steady route between them.

Ad-hoc routing protocols can be categorized as into two types; “proactive” or “On Demand (reactive)”. Proactive conventions demand hubs in a remote specially appointed system to monitor routes to every single possible destination. This is significant in light of the fact that, at whatever point a bundle solicitation to be sent, the routes in advance recognized and can be utilized straight away. At whatever point there's adjustment in the topology, it will be scattered all through the whole system. On-request (reactive) conventions will construct the courses when required by the source hub, all together for the system topology to be identified as required (on-request). At the point when a hub needs to send parcels to destinations however have no routes to the goal, it will begin a route recognition process inside the system. At the point when a route is remembered, it will be supported by a route maintenance procedure until the goal winds up inaccessible or till the route isn't needed any longer. Proactive conventions include the advantage that new correspondences with arbitrary destinations experience minimal delay, but experience the drawback of overhead to bring up to date routing information at every nodes. To defeat with this impediment, responsive conventions take on the contrary technique by finding route to a goal just when required. Receptive conventions routinely use less data transmission contrasted with proactive conventions.

1.2 Broadcast Storm Problem

A straight-forward way to deal with message communication is by flooding. A host, on accepting a communicate message, has the commitment to rebroadcast the message . Unmistakably, this outlays ‘n’ transmissions in a system of it has. In a CSMA / CA organize, downsides of flooding comprise:

Repetitive rebroadcasts: A portable host chooses to rebroadcast a communicate message to neighbors, every one of its neighbors as of now have the message.

Contention: After a versatile host communicates a message, if a considerable lot of its neighbors choose to rebroadcast the message , these transmissions may seriously contend with one another.

Collision: On account of the inadequacy of backoff system, absence of RTS / CTS discourse, & CD, collisions are expected to occur & cause more harm.

1.2.1 Examination on Redundant Rebroadcasts

The accompanying investigation demonstrates that rebroadcasts are very exorbitant and ought to be utilized with alert. Think about the straightforward situation where node A sends a message & host B chooses to rebroadcast the message . Let S_A & S_B signify the circle regions secured by A's & B's transmissions, separately. The extra territory that can profit by B's rebroadcast is S_{B-A}.

1.2.2. Examination on Contention

To address conflict issue, consider the circumstance where node A transmits a message & there are ‘n’ nodes to hear that message. In the event that every one of these hosts endeavor to rebroadcast the message , dispute may happen in light of the fact that at least two host around A are probably going to be close and in this way fight with one another on the remote medium. How about we investigate the less complex instance of n = 2. Let node B & C be the two receivers node. Let B arbitrarily situate at A's transmission range. With the end goal for C to battle with B, it must situate in the region $S_{A \cap B}$. So the likelihood of dispute is $| S_{A \cap B} | / \pi r^2$.

1.2.3. Investigation on Collision

Consider the situation where a few neighbor has hear a communicated from host X. There are a few purposes behind impacts to happen. To begin with, if the encompassing mechanism of X has been tranquil for enough long, every one of X's neighbors may have passed their backoff systems. In this manner, subsequent to hearing the communicate message (and having passed the DIFS time frame), they may all begin rebroadcasting at around a similar time. Second, when impact happens, without collision detection (CD), a host will continue transmitting the bundle regardless of whether some of previous bits have been distorted. Furthermore, the more drawn out the bundle is, the more the waste. Message broadcasting is a vital function in wireless Ad-hoc network in which a node passes on a message “m” to every neighbor thus cause redundant broadcast which is termed as Broadcast storm Problem where every node will be obligated to re-broadcast data packet each time it gets the packet [1][2] . In MANETs, flooding of messages will bring about numerous repetitive correspondences.

Figure 1.1 demonstrates a topology of a MANET. At the point, node “u” broadcasts the packet, node “v” & node “w” receives the packet, At that point, node “v” & node “w” will re-broadcast the packet to each-other. Misleadingly the 2 (two) communications may bring thoughtful broadcast storm problem, where these redundant packet cause collision & contention.

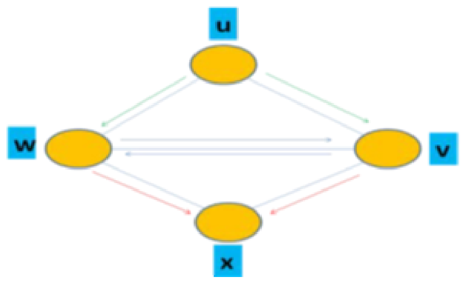

Figure 1.1: Flooding in MANET

In figure 1.2, Bits m_1 and m_2 need to be transferred to both receivers R_1 and R_2. Every link transmits only a bit. Message m_1 and m_2 can be received either on the right or on the left side .

The task is to pick out tiny set of forwarding nodes within the deficiency of global network information. The researchers have done substantial work to seek out Connecting Dominating Set two ways, namely 2-hop & 1-hop neighbor information [3,4]. Numerous broadcast algorithms besides flooding have been proposed [5-11]. Typical global [12] [13] and quasi-global [14] broadcast protocols use either partial global or global information to consensus a small forward node set. To reduce effect of the broadcast storm problem, we should prevent redundant retransmits of the broadcast packet and differentiate the timing of retransmits. Ensuing this recommendation, numerous schemes, called the location-based, distance -based and counter-based were derived.

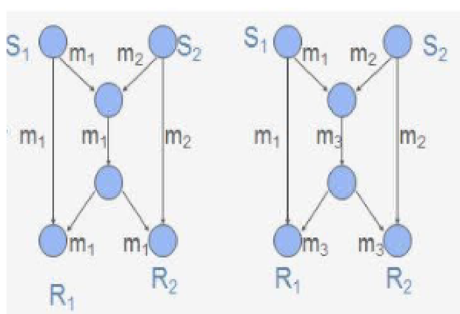

Figure 1.2: Butterfly Network; Source S_1 and S_2 multicast m_1 and m_2 to both receivers

1.3. Introduction to Network Coding

The system of nodes engaged with transmission is spoken to by coordinated diagram G (V, E) where ‘E’ is the arrangement of edges & ‘V’ is arrangement of vertices. The model expect a particular of multicast prerequisite as (R, h, G) where R is the acceptable coding district, h is the data rate in bits per unit time. The system has a source hub S. The source is expected to exchange information to the multicast collectors D and E. The hubs A, B and C are the moderate hubs. Each edge is accepted to have a limit of 1bit. Think about that the source transmits two single bits b1 and b2 to the multicast collectors. This should be possible in two modes m0 and m1 in particular without Network Coding and with Network coding, individually. In mode m0 source S multicasts bits b1 and b2 to the halfway hubs A and B. At that point, the bits are communicated from A and B. The halfway hub C gets both b1 and b2 as appeared in Figure 1.6(a). It communicates b1 and b2 one by one. By this mode m0 the derivations made are that there is a repetitive gathering of bits at the collectors.

In the mode m1, the source S multicast 2 bits to A and B and they in turn communicate the bits. In any case, the hub C broadens its usefulness post routing. It completes a basic encoding capacity, Ex-OR the bits b1 and b2. This broadened conduct of C is spoken to by a spurious hub C' as appeared Figure 1.6(b). Subsequently, in this mode, hub C utilizes Network coding and the derivations are that there is no excess gathering of bits at the collectors.

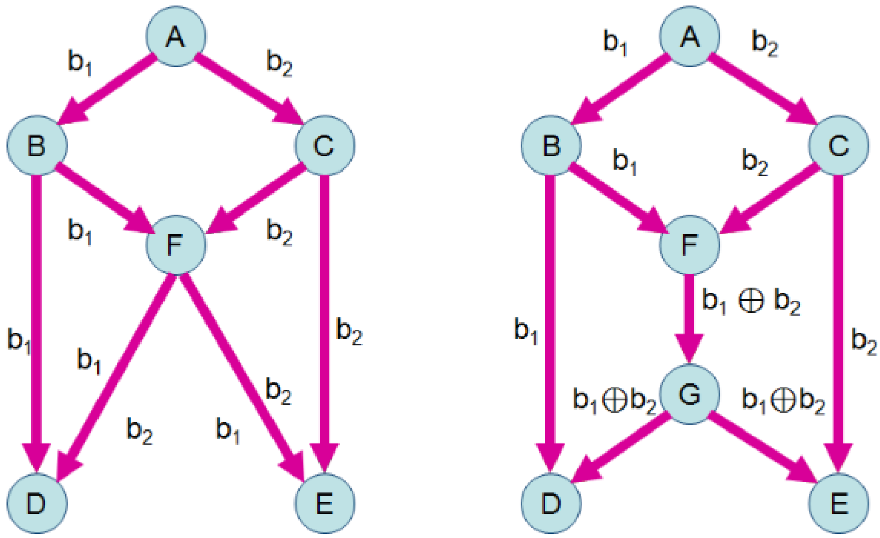

Figure 1.3: Single source multicast network

In this way, with Network coding the required number of transmissions are less. This at last gives increment in throughput. The outcomes are pertinent to the precedent where b_1 and b_2 are two parcels of solitary information rather than two bits of a parcel.

The above precedent is the base model for the diverse aftereffects of many Network coding central research works. All the above outcomes accept that hub C is the encoding hub which brings Network coding into the system. In spite of the fact that every one of the hubs of our application can possibly encode the information bundles, they require not generally do the Network coding. Along these lines, in Figure 1.6 of the three sending hubs (A,B,C), C alone is expected as an encoding hub.

1.4 Issues in Ad-hoc Network Using Network Coding

To construct a protocol which conveys the above advantage, we need to concentrate on few challenges.

1. **How Many Packets to propel:** Node keeps sending a packet until next-hop obtains it or until it gives up in traditional best path routing. There is no particular next-hop; all nodes nearer to the destination than the current transmitter can participate in forwarding the packet with opportunistic routing. How many transmissions are adequate to make sure that at least one node nearer to destination has received packet?

2. **Stop and Purge:** Routers send linear combinations of packets with network coding. Once the destination has heard adequate such coded packets, it decodes & repossess the file. We need to stop sender once destination has obtained the transfer & purge related data from the forwarders .

3. **Efficient Coding:** Network coding optimizes for better utilization of wireless medium, but coding necessitates the routers to multiply & add the data bytes in packets. We need efficient coding & decoding strategies to avoid routers' CPU from a tailback .

1.5. Design Principles

1. To fabricate the Network Coding plan, we need to settle on a couple of structure choices. In the first place, we structure our Network Coding plan around the guideline of never deferring packets. At the point when remote channel is accessible, hub takes the bundle at the leader of its yield line, checks which different packets in the line might be encoded with this packet, XORs those packets together & communicates the XORed form.
2. Second, offer XORing parcels of similar lengths, in light of the fact that XORing smaller bundles with larger ones decreases transmission bandwidth savings.
3. Third, Searching for proper packets to Network Coding is productive because of the support of virtual lines.
4. At last, we need to guarantee that each neighbor to whom a Network Coded packet is going has high likelihood of unraveling its native packet.

1.6. Quasi Global and Total Global Protocol

As of late, various research bunches have master presented progressively effective broadcasting methods [15-21] with different objectives, for example, limiting quantity of retransmissions, limiting total control utilized by transmitting hubs, limiting general postponement of broadcasting, etc. Camp and Williams [22] grouped communication protocols in four classifications: basic (daze) flooding, likelihood based, territory based, and neighbor knowledge techniques. Yang and Wu [23] grouped telecom conventions dependent on neighbor learning data: local, quasi – local, quasi- global, global.

In local broadcasting, a circulated communicate convention depends on exclusively nearby state data. All conventions that select forward hubs privately (in light of 1-bounce or 2-jump neighbor set) have a place with this class. It has been perceived that adaptability in remote systems can't be accomplished by depending on arrangements where every hub requires global information about the system. To accomplish versatility, the idea of confined calculations was proposed, as conveyed calculations where straightforward neighborhood hub behavior, in view of nearby learning, accomplishes an global objective.

In quasi-local broadcasting, a dispersed communicate convention depends on for the most part nearby state data and infrequently partial global state information . In case of quasi-global broadcasting, broadcast protocol is dependent on partial global state information. For instance, estimation algorithm in depends on building global spanning that is developed in a succession of consecutive proliferations.

The distributed or centralized (global broadcast protocol), is based on global state data. In this part, we classify recently proposed broadcasting conventions into a few families: centralized methods, distributed methods and localized methods . Centralized methods compute a tree utilized for broadcasting with different optimization of tree. In case of localized methods, every hub needs to fundamentally maintain the condition of its nearby neighbors (inside a few consistent jumps). In the wake of getting a bundles that required to be handed-off, the hub chooses whether to transfer the bundle just dependent on its nearby neighborhood information. Dominant parts of the conventions are in this family.

In case of Distributed techniques, hub may require few information in excess of a consistent jump away to choose whether to transfer the message. For instance, communicating based on MST developed in a dispersed way is a distributed strategy, yet not localized technique as we can't build Minumum Spanning Tree in a localized way. Researchers effectively explained a few testing questions [24-30] by giving efficient limited calculations.

1.7. Cluster & Dominating Sets Theory

The compelling use of the intensity of the Ad Hoc organizes lies in adaptability, strength, viability and quick combination of these systems. The route discovery & route development in the Ad Hoc systems ought to be done quickly. Performing successful course discovery on such situations can be best accomplished when the telecom is made strides further more, the excess in course location is evacuated.

The Clustering calculation frames disjoint legitimate gatherings of hubs with a lead hub as cluster head of the bunch. cluster head takes part in the steering and has extra errands of group upkeep. The clusters frame a sensible backbone and the Route seek space is constrained to the number of clusters. Be that as it may, the technique shapes suboptimal routes.

Dominating set (DS) has been generally utilized in the choice procedure of a functioning hub set. A set is dominating if each hub in system is either in the set or a neighbor of a hub in the set. At the point when dynamic hubs shape a dominating set, all hubs in the system are additionally said to be reachable. At the point when a DS is associated, it is indicated as a CDS; that is, any two hubs in the DS can be associated through transitional hubs from the DS. CDS as an associated virtual backbone has been generally utilized for communication process, looking in a decreased space. As a result of the unbridled accepting method of remote sensors, when every hub in a CDS advances the parcel once, all hubs in the system will get the bundle. In Figure. 1.4 (a), the dominating hub set (w, v, u) shapes a virtual backbone for proficient broadcasting, since just dominating hubs require to forward the bundle and every residual hub can get the parcel without sending it.

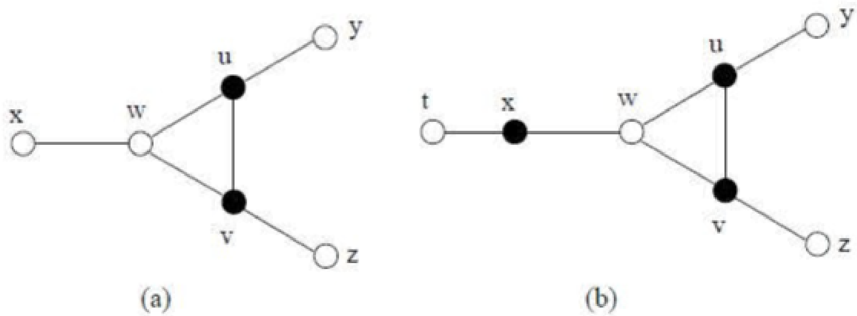

Figure 1.4: (a) A sample network: (w, v, u) forms a CDS and (v,u) forms an ECDS. (b) Another sample: (v,u,x) forms an EWCDS. [Jie Wu, Mihaela Cardei, Fei Dai, and Shuhui Yang]

The Connected Dominating Set based calculation frames a sensible backbone of associated gateways. The communicate bundles are retransmitted by these hubs. The routes are figured by shortest path calculation. Changes in the backbone does not have affect on the progressing interchanges. New routes are computed upon failure of existing routes. The efficiency of a correspondence path depends not just on its control conventions, yet in addition on its topology. Every hub decides the participation in the MDS for itself and its one-jump neighbors dependent on two-bounce neighbor data that is spread among neighboring hubs. The calculation at that point guarantees that the individuals of the MDS are associated into an associated commanding set (Cds), which can be utilized to frame the spine infrastructure of the correspondence network for such purposes as routing.

In these MANETs, clustering is a standout amongst the most essential ways to deal with energy proficiency and cost effective correspondences. clustering is an calculation in which the system is isolated into non-covering sub systems, alluded to as clusters where each hub of each sub arrange is at the most k-bounces from a recognized station called the cluster head (CH). Clustering is a progressive structure, and accordingly is reasonable for a generally huge numbers of hubs [31, 32, 33, 34]. Clustering is directed by first choosing Cluster-heads. Non-cluster h heads pick clusters to join and after that progress toward becoming individuals. In spite of the fact that there are a few sorts of clustering calculations, widely used algorithm is lowest ID algorithm.

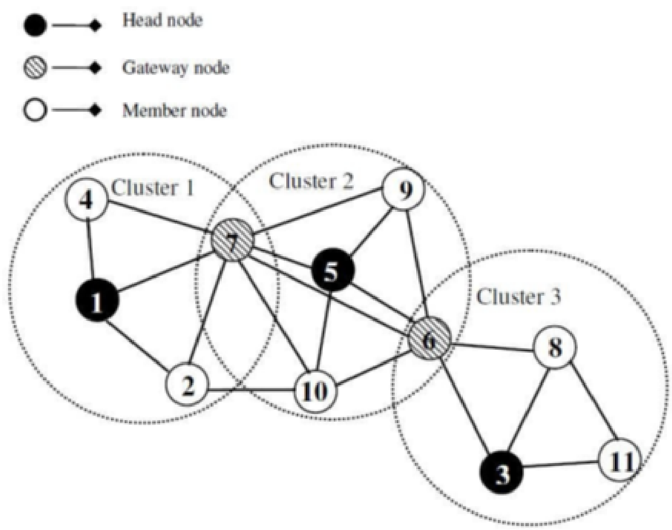

Figure 1.5: Cluster formation using lowest ID clustering algorithm [A. Vasudeva and M. Sood],

In the lowest-ID clustering algorithm as per figure 1.5 a hub with the least ID is picked as cluster head. Every hub is furnished with a remarkable ID & it occasionally communicates rundown of its neighbor's IDs, including itself. A hub which just hears hubs with ID higher than itself is a cluster head (CH). The most minimal ID hub that a hub hears is its clusterhead, except if the least ID explicitly surrenders its job as a clusterhead when a hub with a lower ID goes into a similar group. This is a basic calculation and the procedure is quick. Likewise, the rate of progress of clusterhead is low and thus the framework execution is better as far as throughput. On the other hand, the quantity of clusters may turn out to be unfortunately high because of which the bundle conveyance deferral may wind up over the top. Besides, clusterheads with littler IDs experience the ill effects of the battery seepage, bringing about short lifetime of the framework. Figure 3 demonstrates a schematic of the aftereffect of utilizing most minimal ID bunching. There are eleven hubs with one of a kind IDs, which shape a associated graph. After the Lowest ID grouping calculation is executed, three bunches are framed, as portrayed by the specked circles. The dark shaded balls inside each bunch speak to the cluster heads (5,3 & 1 in figure 3). The striped balls (6 & 7) that are inside the correspondence scope of at least two distinct clusters speak to the entryway hubs and the void balls are the member hubs.

The manner in which Network coding is utilized can likewise be grouped into three principle classifications relying upon the kind of blends; those groupings are: Joint Inter and Intra-Session Network Coding; Inter-session Network coding; & Intra-session Network Coding [35, 36].

When we code the bundles from a similar source, we are performing intra-session arrange coding. Inter session Network Coding can lessen the quantity of transmissions when the parcels from distinctive source are coded together. On the off chance that the remote medium have lossy connections it is best to have between session and intra-session organize coding together as joint process. Identifying parcels and registering the cardinality of coded messages can be effectively done utilizing address bit vectors.

1.8. Problem Statement

After looking through the current research progress of solution of broadcast storm problem and network coding, the challenges which we are facing in the wireless environment are solidified to:

1. How do we propose a generalized coding condition to minimize number of forwarding nodes performing network coding of packets together?
2. How do we propose an effective coding scheme which is robust to provide the transmission reliability while keeping minimum redundancy?

The aim of this work is to achieve lesser redundancy in terms of achieving less number of forwarding nodes for retransmission using the concept of Network Coding to subtract the effects of Broadcast Storm Problem. So in our work, we have proposed various algorithms as follows:

Reduction in Number of Forwarding Nodes for Network Coding (NC)

1. By Considering Global Network Information and Probability of transmission of packets.

2. Under different circumstances of uneven incoming traffic and outgoing traffic degree of nodes along with other attributes like of input and output bandwidth.
3. Using Address Bit Vector Integrated Network Coding scheme to avoid repeated XOR operation between similar packets.
4. By calculating correct Inference of nodes under topology SET.
5. Using AODV protocol for route discovery to reduce sub-graph.

Via simulations, we show the promising results. Details of our contributions will be lined out in next chapters.

1.9. Outline of the Thesis

The thesis has been organized into 6 chapters. Chapter 1 presents Introduction comprises problem statement and various issues in mobile ad-hoc networks and broadcasting methods.

Chapter 2 presents the existing work to subtract the broadcast storm problem like Counter-based scheme, Location-based scheme, Probabilistic scheme, Distance-based scheme, Cluster-based scheme etc. This chapter also discussed the various methods like prim's algorithm and kruskal's algorithm. Chapter 3 presents four algorithms for identification of nodes which will perform Network Coding using Global Network Information and Probability of transmission of a node to reduce number of forwarding nodes. This chapter also includes algorithm which considers incoming traffic degree and outgoing traffic degree of a node along with sum of input and output bandwidth of a node to decide forwarding nodes where network coding needs to be applied.

Chapter 4 presents two algorithms where first algorithm explains the concept of Address Bit Vector Integrated Network Coding and second algorithm address about Inference of topology SET using Network Coding to reduce number of forwarding nodes. Chapter 5 presents an algorithm where AODV protocol is integrated with Network Coding (NC) approach to make maximum utilization of both the theories to reduce number of forwarding nodes . Finally, followed by the conclusion and future scope of the research work for further research are provided in Chapter 6.

CHAPTER 2

LITERATURE REVIEW

In General, once a flooding data packet is received by a node then it elects whether or not to relay it to its neighbor. The Neighbor topology primarily approaches to escape the broadcast redundancy in MANET's. We here determine the minimum number of forwarding node set which forms a minimum connected-dominating-set. It could be set of nodes if each node is either within the set or the neighbor is in this set. The task is to pick out tiny set of forwarding nodes within the deficiency of global network information. The researchers have done substantial work to seek out Connecting Dominating Set two ways, namely 2-hop & 1-hop neighbor information [37, 38].

Numerous broadcasting algorithms in addition to blind flooding have been advised [39-45, 85,101-109]. Typical global [46, 47] & quasi-global [48] broadcast protocols use either partial global or global information to consensus a small set of forward node. To reduce effect of the broadcast storm problem, we should prevent redundant retransmits of the broadcast packet and differentiate the timing of retransmits. Ensuing this recommendation, numerous schemes, called the location-based, distance -based and counter-based were derived.

They are dependent on many mechanisms to support a host to assert the redundancy of a rebroadcast and choose whether to rebroadcast or not. Results display that these can efficiently reduce the side effects of broadcast storm problem [49].

Figure 2.1 shows the cluster-head based broadcast algorithm.

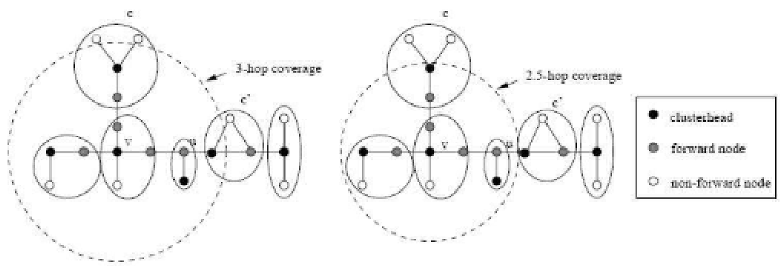

Figure 2.1: Cluster-head based broadcast algorithm

Figure 2.2 shows that removal of node “u” will not eliminate all paths between nodes “x” and “y” [50].

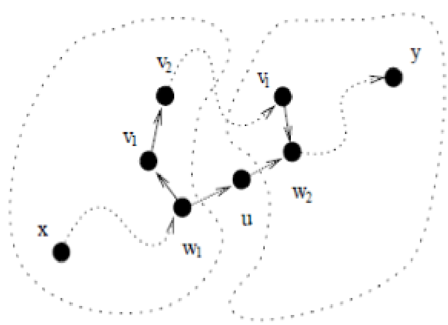

Figure 2.2: Removal of node “u” will not eliminate all paths between nodes “x” and “y”

Figure 2.3 shows the failure of dominant pruning algorithm [51] since node “C” and node “B” are dismissed in 2nd iteration of communication when they already exists in the sender list of node “A”

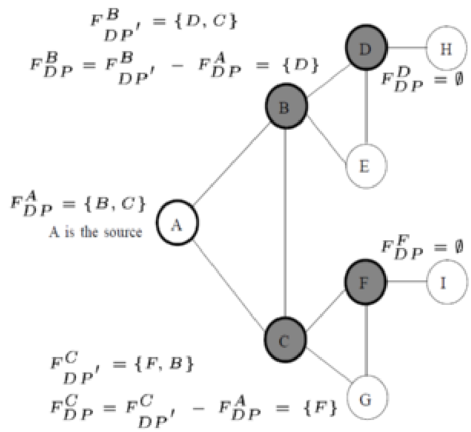

Figure 2.3: Failure of dominant pruning algorithm [Marco Aurelio ´ Spohn , J.J. Garcia-Luna-Aceves]

We begin with a scientific classification of communicate conventions in MANET. This scientific classification fills two needs: (i) to offer a stage of reasonable examinations among conventions under various suspicions and models and (ii) to put our methodology as per the scientific classification so as to plainly express its commitments and constraints. To have an unmistakable center, we restrain our regard for two destinations: (i) forward nodes set (ii) low overhead.

Our first grouping depends on the sort of calculation utilized: deterministic or probabilistic. The probabilistic approach [48, 52] for the most part meets the two targets without presenting much overhead. Be that as it may, with a low likelihood, at least one basic properties may not be met. For instance, the created DS is not ensured to be associated as in [48] or the system isn't ensured to be secured by the chosen forward hubs as in [52].

Our second characterization depends on the sum of state data utilized in the calculation: global or neighborhood. Through a few rounds of successive data trades, global data can be collected based on neighborhood data as it were. Four kinds of communicate conventions are examined based on the second characterization:

1. **Local:** It is based on exclusively nearby state data.

2. **Quasi Local:** It is based on most part of neighborhood state data and intermittent global state data.

3. **Global:** It is dependent upon global state data.

4. **Quasi Global:** It is based on partial global state data.

One normal misinterpretation is about the distinction between the centralized algorithm where one hub, say the source hub, decides the total communicate process & the distributed algorithm where the communicate process is resolved appropriated. Likewise, any model with nearby data can be actualized in an incorporated way once adequate measure of data is gathered at the source through data trades. To give a uniform model, we expect that at first every hub knows just its neighbor set, partial global or global state information is gathered through data trades (as far as rounds) among neighboring hubs.

Dissimilar to global broadcast protocol, the quasi-global communicate convention does not have to gather entire global state. Just partial global state data is gathered. Convention proposed in [39, 53] fits into this class: Spanning tree is first built beginning from chosen root.

Already proposed several heuristics to reduce rebroadcasts are:

1. **Probabilistic scheme:** Randomly chosen probability is the deciding factor for Rebroadcast the packet.
2. **Cluster based scheme :** Gateways & cluster heads forward again
3. **Location based scheme:** Pre-acquired location information of neighbors is the deciding factor for Rebroadcast the packet.
4. **Counter based scheme:** Threshold value of counter decides whether to re-broadcast or not.
5. **Distance based scheme :** Relative distance between hosts to make decision of rebroadcasting

Network coding can be ordered into 3 primary classes relying upon the sort of combinations; which are: Inter-session NC Intra- session NC, & Joint Inter & Intra Session NC. Inter Session

NC can decrease the number of transfers when packets from distinctive source are network-coded. We are coding the packets from a similar source, as doing intra-session NC.

Using directional antennas the effect of broadcast storm can be alleviated. This is illustrated using table 2.1 and table 2.2 for network sample illustrated in figure 2.4

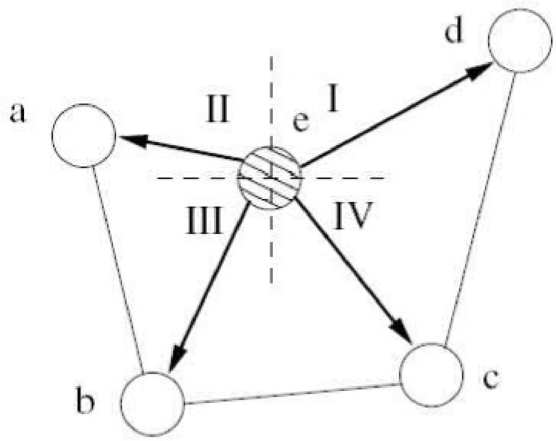

Figure 2.4: A sample network

Table 2.1 shows Neighbour reception table of node "e" for figure 2.4 and table 2.2 shows Transmission of node 'e' using directional antennas and Network coding.

Table 2.1: Neighbour reception table of node e

Neighbor reception	**A**	**B**	**C**	**D**
e	1	1	1	1
d	0	0	1	1
c	0	1	1	1
b	1	1	1	0
a	1	1	0	0

Table 2.2: Transmission of node 'e' using directional antennas & Network Coding

Transmission	P_1	P_2
IV	1	0
III	0	1
II	1	1
I	1	1

In Figure 2.4, P1 need not to be transmitted in sector III. P2 need not to be transmitted in sector IV. Cost is 16 when e transmits all 4 messages omni-directionally. Cost is 8 when Network Coding is used. Cost is 6 when both directional antennas concept and Network Coding is applied.

Kruskal's algorithm:

```
En-queue edges of Graph in queue in growing order of cost.
'T' = ϕ ;
while(queue ≠ empty)

{
  de-queue edge' e';
  if('e' does not form a cycle in 'T')
     add 'e' to' T';
}
Return 'T';
```

Prim's algorithm uncovers a minimum cost spanning tree by picking edges from the graph one-by-one as follows :

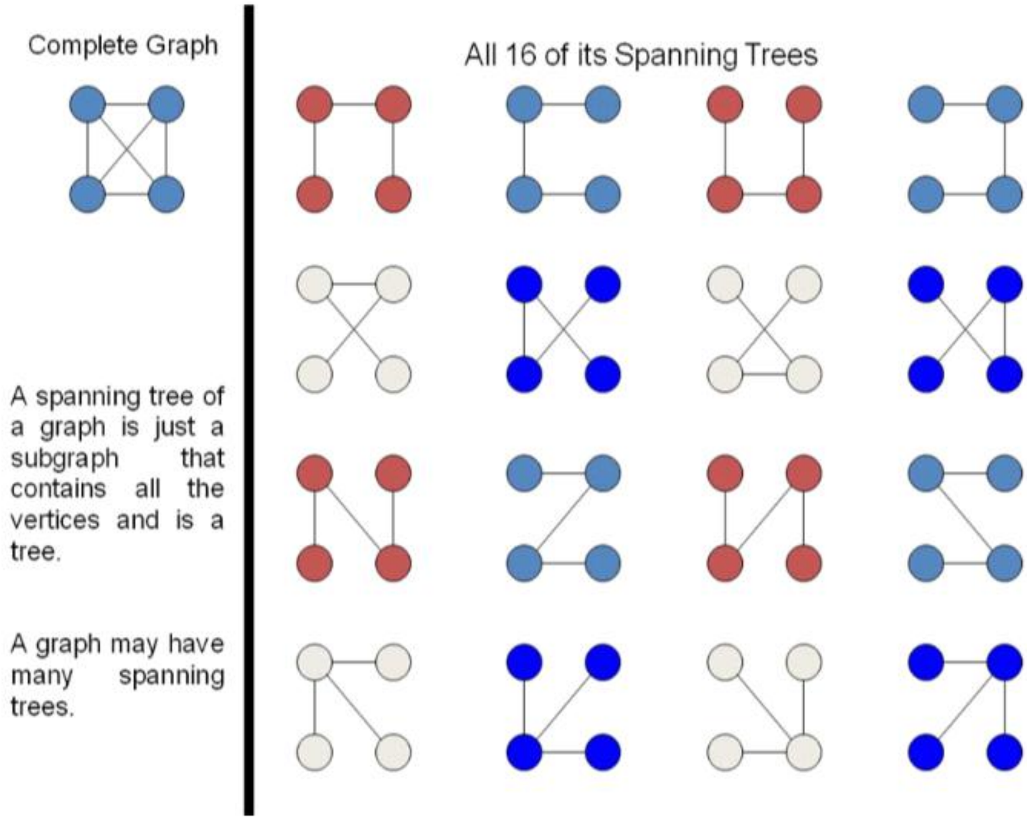

Figure 2.5: Illustration for finding all possible set of spanning tree for a given graph

Prim's algorithm:

Graph G=(V, E);

Let T be tree consisting of the preliminary vertex '**x';**

while (T has < than IVI vertices)

{

locate minimum edge involving T to (G-T);

adjoin it to T;

}

Figure 2.5 shows the illustration for finding all possible set of spanning tree for a given graph which is further used as a subgraph. The issue of broadcast support in MANETs has been broadly examined in [54-57]. In [54], the issue of high overhead of utilizing basic flooding to sustain broadcast was focused. The issue of least enery consumption broadcasting has been observed to be NP-complete [59] and from that point forward an extensive number of estimation calculations have been proposed. These are either deterministic [54, 5] or probabilistic ways to deal broadcast effectively. In probabilistic calculations, packetss are just sent with a specific likelihood [55, 57,5,92,97]. In deterministic methodologies, if entire topography data is known, will give ideal outcomes[59]. However, where topology continues evolving much of the time, it's very impractical to get complete topology.

1. COPE

To suggest the researchers a vibe for how COPE functions, we begin with a genuinely straightforward case. Take the situation in Figure. 2.6, where Alice & Bob need to swap over a set of packets. In available present methods, Alice sends the data packet to connecting router, which further send that packet to Bob, & Bob forwards this to the connecting router, which further forward this to Alice. This procedure needs four number of transmissions. Presently following a network coding (NC) standard, both of them transmit their separate packets to connecting router, which apply XOR to the both packets and shows mixed form. It takes three number of transmissions rather than four as per figure 2.7 & 2.8. Spared transmissions can be utilized for broadcasting distinct information.

COPE uses 3 principle approaches:

(i) **Opportunistic Listening:** Wireless systems provide numerous open doors for mobile nodes to hear packets with omni-directional antenna. Furthermore, every node telecasts received reports to convey its neighbor which data packets this has put away.

(ii) **Opportunistic Coding:** To achieve the gain in throughput, the main question that stands is which packets are to be network coded together. A node may have numerous alternatives, however its objective is to boost the quantity of local packets conveyed, guaranteeing that every proposed next-hop has sufficient data to translate its local packet.

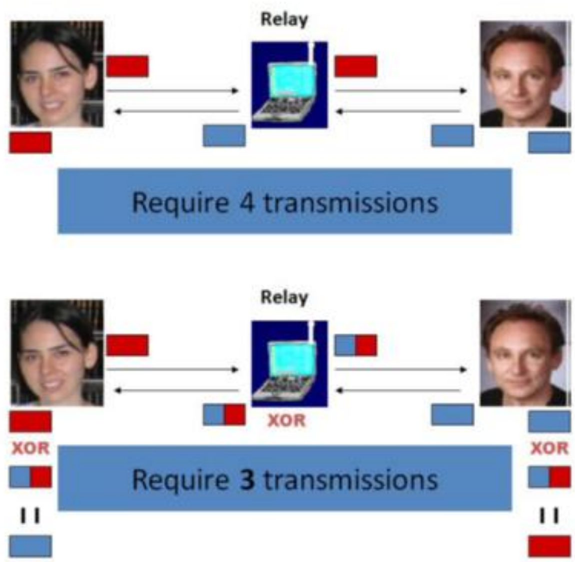

Figure 2.6: Throughput gain using COPE

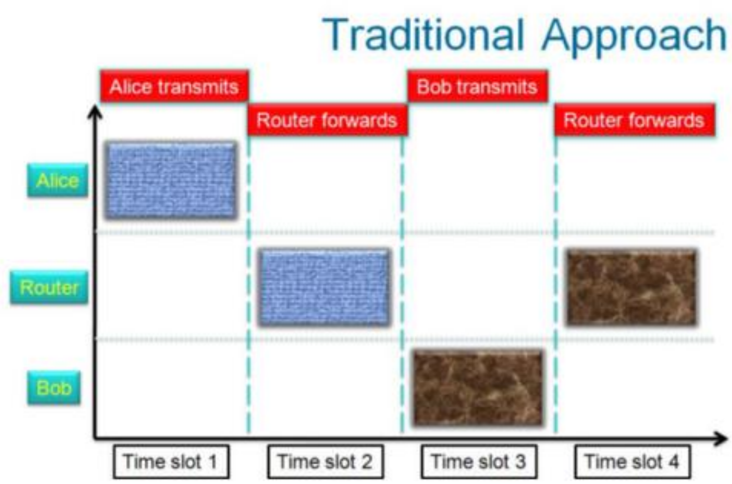

Figure 2.7: Traditional Broadcasting Approach

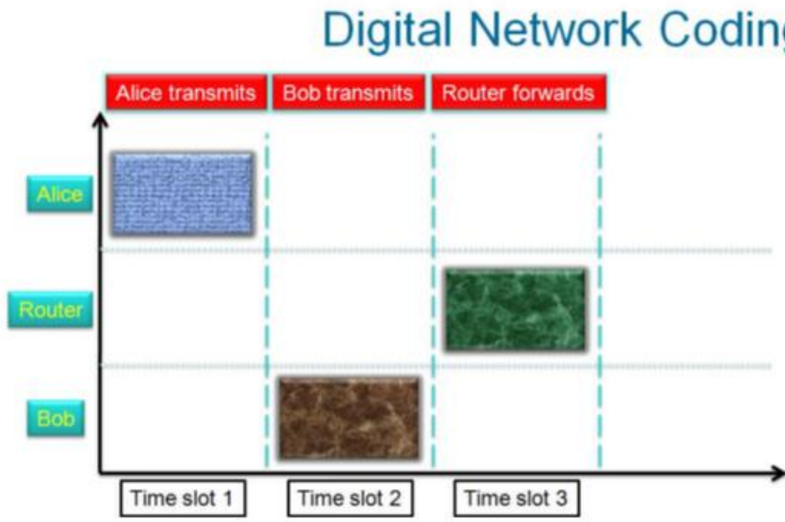

Figure 2.8: Digital Network Coding

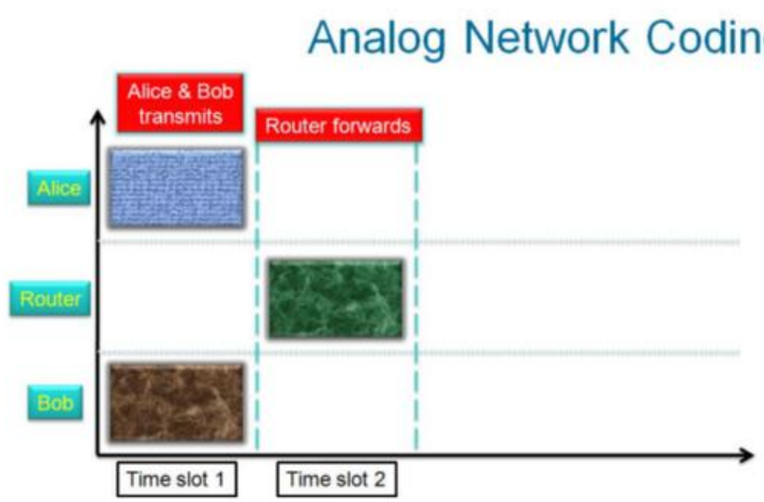

Figure 2.9: Analog Netwok Coding

The coding calculation ought to guarantee that all nexthops of an XORed packet can interpret their comparing local packets [58]. This can be accomplished utilizing the below given straightforward principle

To deliver 'w' packets, to 'm' nexthops, a node can mix the 'w' packets together just assuming each beneficiary has (w – 1) packets .

(iii) **Learning Neighbor State:** But how to recognize what packets a node's neighbors have? As clarified before, every node pronounces to its neighbors the packets it keeps. Consequently, a node can't depend exclusively on receiving reception reports [58].

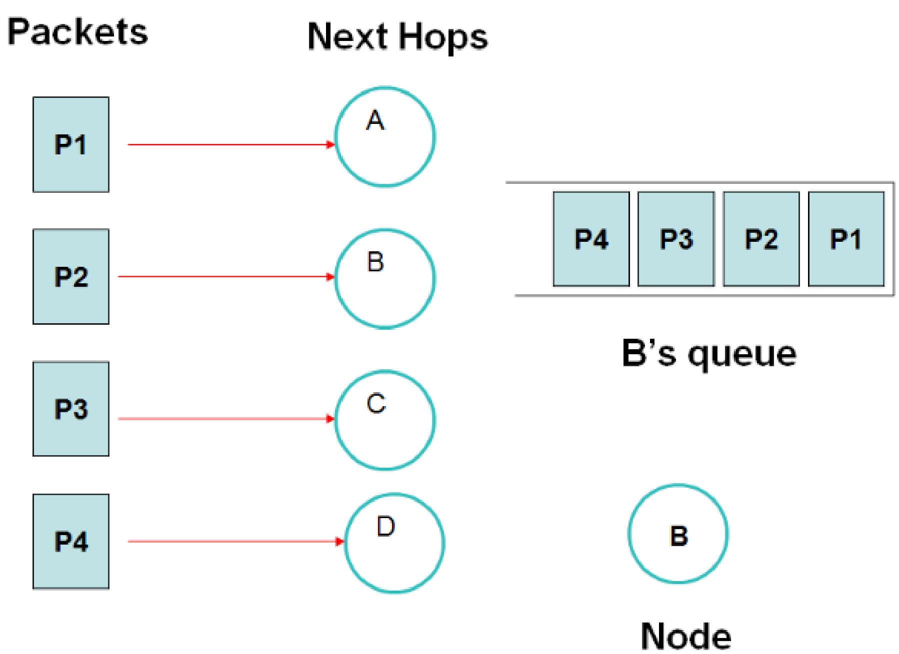

Figure 2.10: Next-hops of packets in B's queue

Figure 2.10, 2.11, 2.12 & 2.13 point outs simple rule for choosing which packets to code together.

COPE's throughput gain widens as the possibility of network coding raises. At the point when there is higher movement, more packets are accessible at the moderate nodes, and thus there are all the more network coding possibilities.

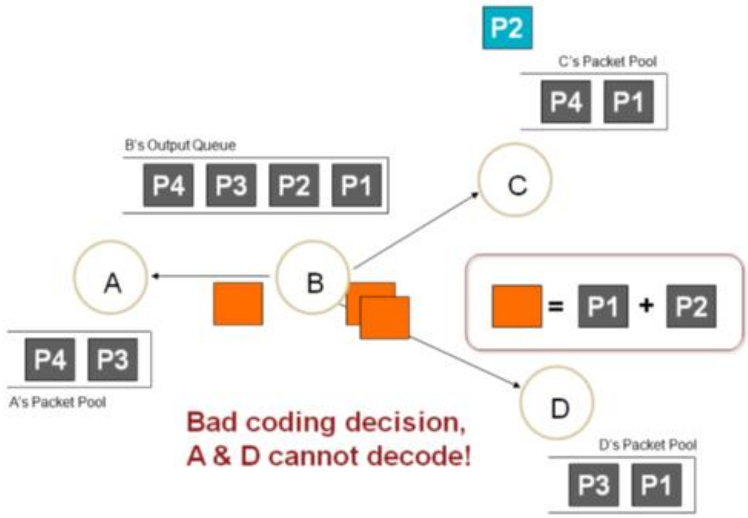

Figure 2.11: Bad Network coding Decision

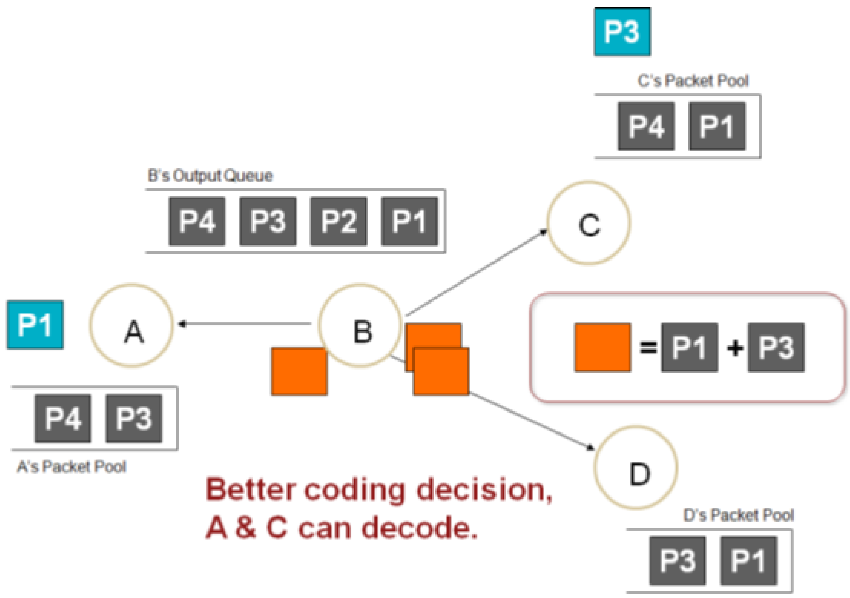

Figure 2.12: Improved Network coding Determination

Measurement metrics:

- **Network Throughput:** the deliberate aggregate end - 2 - end throughput
- **Throughput Gain:** the proportion of the network system throughputs with COPE & exclusive of the COPE.

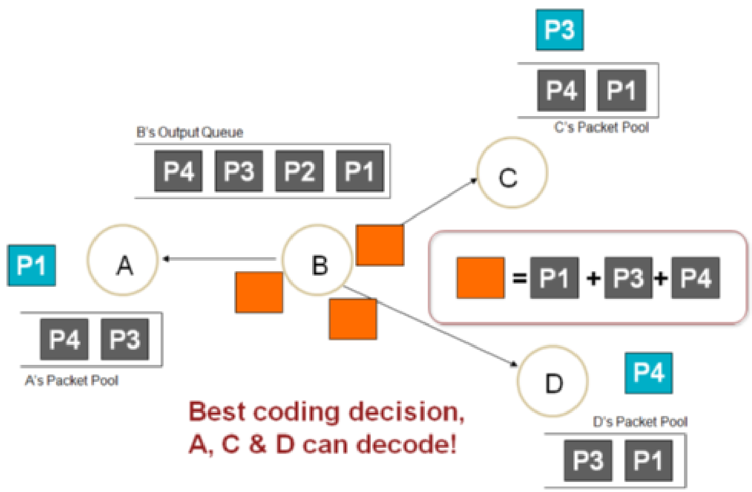

Figure 2.13: Best Network coding Determination

The significance of packets combination using network coding was a fundamental subject for a large portion of the specialists in the previous couple of years. The principle center of system configuration has been founded on effortlessness of operation & versatility. However, late works demonstrates additionally that viable power sparing and physical asset administration should be possible by communicating with the different schemes of NC. Hui and Culler[61] presented Deluge a dependable information spread convention to beat the propagation of huge information objects that can hear it.

However an extensive new convention was presented in [62] where 40% less packets were utilized as a part of vast systems than Deluge and load adjusting was additionally mulled over despite the fact that it was not obviously checked in the algorithm. AdapCode used to disperse information in the system utilizing NC (Network coding) [63] yet in a versatile route as indicated by link quality. To concentrate on information accumulation algorithms, Wang et al. [64] presented another scheme in NC called Partial Network Coding (PNC) attempting to conquer the test of consistent information accumulation from sensors to server where information expulsion was accessible. SenseCode [60] was presented as an accumulation convention in wireless networks utilizing NC as many two one correspondence convention in which the time was separated into rounds.

CHAPTER 3

REDUCING THE NUMBER OF FORWARD NODES FROM 1-HOP NODES TO COVER 2-HOP NODES WITH NETWORK CODING

3.1. Introduction

Wireless medium is creating numerous fortuities for nodes to obtain packets when they are not the projected recipient. MANET is specially appointed Network that getting extraordinary conspicuousness inside the general public. Message broadcasting is a basic function in wireless Ad-hoc network in which a node passes on a message "m" to all neighbors thus causing redundant broadcast which is called as "Broadcast storm Problem (BSP)" in which every node will be obligated to re-broadcast data packet every time it gets the data packet for the 1st time [67, 66]. In MANETs, flooding of messages will bring about numerous repetitive correspondences. Figure 3.1 demonstrates a topology of a MANET. At the point, node "u" broadcasts a packet, node "v" & node "w" receives packet, At that point, node "v" & node "w" will re-broadcast packet to each other.

Misleadingly the two communications may bring thoughtful broadcast storm problem, where these redundant packet cause collision & contention.
In a CSMA/CA network, disadvantages of flooding are:

1. **Redundant rebroadcasts-** Host adopts to re-broadcast a message "m" to its neighbors, they previously have that broadcast message "m".

2. **Contention-** Later mobile host communicates message "m", if large number of its neighbors choose to re-broadcast message "m", these broadcasts may ruthlessly contend.

3. **Collision-** They are more probable to happen and to produce more harm because of the inadequacy of back-off mechanism, the non-existence of CD and the absence of RTS/CTS dialogue.

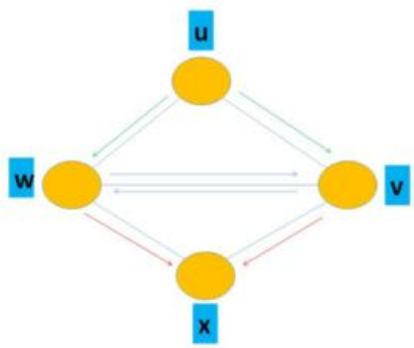

Figure 3.1: Flooding in MANET

In figure 3.2 (a), Bits m_1 and m_2 need to be transferred to both receivers R_1 and R_2. Every link transmits only a bit. Message m_1 and m_2 can be received either on the right or on the left side.
Solution: Compute XOR (i.e. Apply network coding) in the middle link (Figure 3.2 (b)) and both sides get m_1 and m_2. Table 3.1 clearly demonstrates this. Intermediate nodes will further send packets which are XORed of previous received bits [68].

Table 3.1 XOR operation between m1 and m2

m2	m1	m3= m2 ⊕ m1	m2= m3 ⊕ m1	m1= m3⊕ m2
1	1	0	1	1
0	1	1	0	1
1	0	1	1	0
0	0	0	0	0

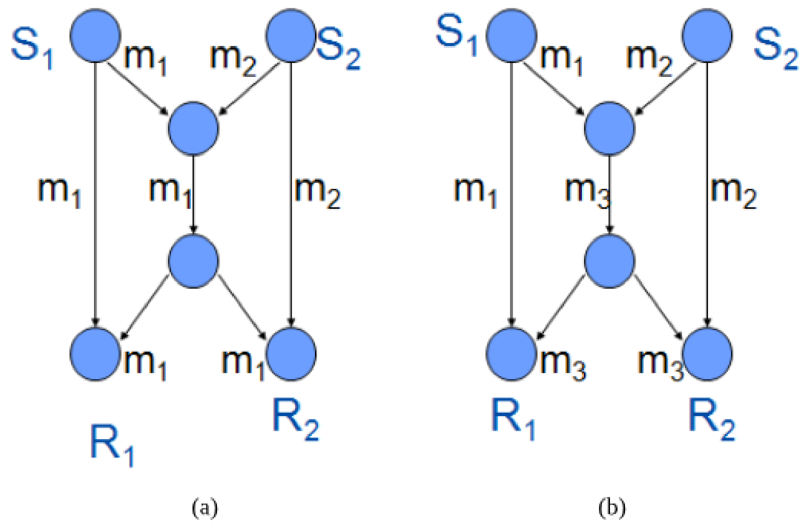

Figure 3.2: Butterfly Network; Source S_1 and S_2 multicast m_1 and m_2 to both receivers

3.2. Proposed Algorithm

The main task of the proposed algorithm is to identify the nodes which will perform forwarding of the network coded packets and at the same time the number of such kind of nodes has to be get reduced and how to perform this based on local information that to be without consulting with rest of the nodes.

Figure 3.3 shows the elimination of node 'E and 'F' from broadcast list of node A since they got similar data bit or message from neighbor B & C. Such kind of nodes needs to be eliminated out to reduce the number of forwarding nodes. Parameters are:

Single path forwarding - Only one path is employed in order to forward traffic to routed destination [69].

Multicopy- Each packet is replicated on all available paths employing in this way the maximum possible redundancy

Multipath - Each packet is assigned on a specific path with different packets of a flow being assigned on different paths. It employs zero redundancy.

Table 3.2 shows the simulation environment parameters which are considered.

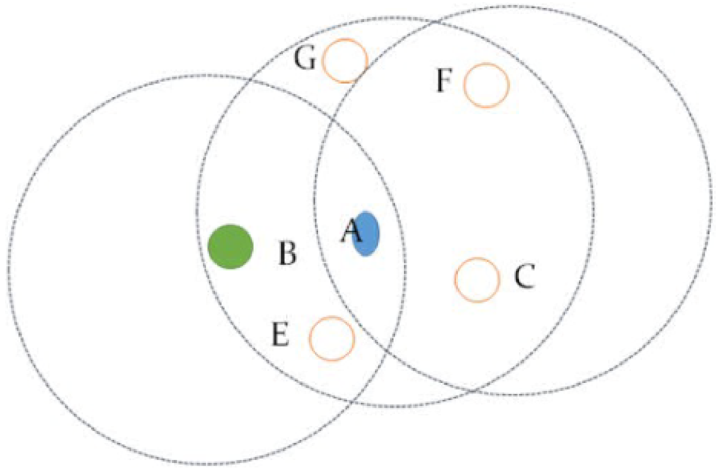

Figure 3.3: Elimination of neighbor node E and F from the list of Node A.

Table 3.2: Simulation Environment

Simulation Parameter	**Value**
Tool Used	NS - 2
Topological size	900m * 900m
Range of Transmission (m)	300
Bandwidth (Mbps)	2.5
Speed (m/s) – Max	5.2
Speed (m/s) – Min	1.2
Pause time	0 s
Packet Rate (packets/sec)	4
Packet size (bytes)	512
Number of CBR Connection	15
Traffic Type	Constant Bit Rate
Interface Queue Length	60

Figure 3.4 is considered for the explanation purpose of our algorithm.

In Figure 3.4,

s = Sender (Node 6)

r = Receiver

N (r) = neighbors of node v

N(N(r)) = neighbors of N (r)

U(s , r) = 2-hop neighbor set

F (s , r) is the forward node list

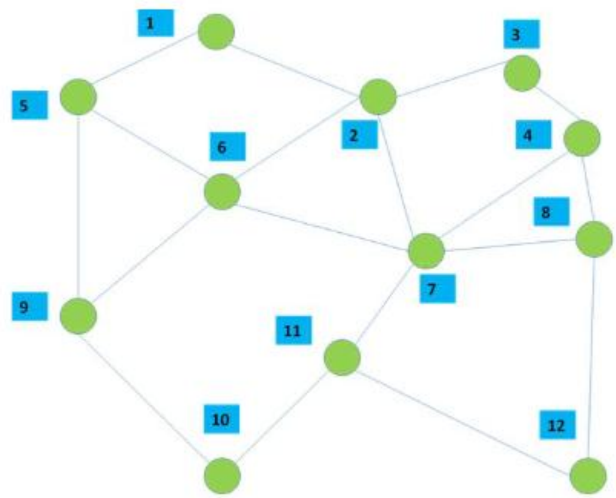

Figure 3.4: Network of twelve nodes with node 6 as source node

Table 3.3 shows the 1-hop and 2-hop neighbor-hood information. It appears to be normal to allow a node with additional number of neighbors transmit prior, as the substantial number of secured nodes will more probable render other booked retransmissions repetitive [65].

Table 3.3: 1-hop and 2-hop neighbor-hood information for Figure 3.4 network

R	N(r)	N(N(r))
12	12,11,8	12,11,10,8,7,4
11	12,11,10,7	12,11,10,9,8,7,6,4,2
10	11,10,9	12,11,10,9,7,6,5

9	10,9,6,5	11,10,9,7,6,5,2,1
8	12,8,7,4	12,11,8,7,6,4,3,2
7	11,8,7,6,4,2	12,11,10,9,8,7,6,5,4,3,2,1
6	9,7,6,5,2	11,10,9,8,7,6,5,4,3,2,1
5	9,6,5,1	10,9,7,6,5,2,1
4	8,7,4,3	12,11,8,7,6,4,3,2
3	4,3,2	8,7,6,4,3,2,1
2	7,6,3,2,1	11,9,8,7,6,5,4,3,2,1
1	5,2,1	9,7,6,5,3,2,1

3.2.1. Proposed Algorithm - BSP-NC-ALGO-1

1. U (s, r) = N (N (r)) – N (r) – N (s)
2. B (s, r) = N (r) – N (s)
3. Node r determines F (s, r). (F (s,r) can be chosen from B(s,r) to cover U(s,r)).

Table 3.4 shows the result of BSP-NC-ALGO-1.

Table 3.4: Result of BSP-NC-ALGO-1

s	r	U	B	F
φ	6	11,10,8,4,3,1	9,7,5,2	9,2,7
6	7	12,10,3,1	11,8,4	4,11
6	2	11,8,4	3,1	3
6	9	11,1	10	10
7	11	9	12,10	10
7	4	12	3	[ ]
2	3	8	4	4
9	10	12,7	11	11

Result of BSP-NC-ALGO-1 clearly shows that the total numbers of forward nodes are 8 while total number of nodes is 12.

3.2.2. Proposed Algorithm - BSP-NC-ALGO-2

1. P (s, r) = N (N (s) ∩ N (r))
2. U (s, r) = N (N(r)) – N (r) – N (s) − P (s, r)
3. B (s, r) = N (r) − N(s)
4. Node r determines F (s, r). (F (s,r) can be chosen from B(s,r) to cover U(s,r)).

Table 3.5 shows the result of BSP-NC-ALGO-2.

Table 3.5: Result of BSP-NC-ALGO-2

s	**R**	**P**	**U**	**B**	**F**
φ	6	φ	11,10,8,4,3,1	9,7,5,2	9,2,7
6	7	7,6,3,1	12,10	11,8,4	11
6	2	11,8,6,4,2	Φ	3,1	[ ]
6	9	9,6,1	11	10	10
7	11	φ	9	12,10	10
9	10	φ	12,7	11	11

Result of BSP-NC-ALGO-2 clearly shows that the total numbers of forward nodes are 6 while total number of nodes is 12.

3.2.3. **Proposed Algorithm- BSP-NC-ALGO-3**

1. U (s, r) = N (N (r)) – N (N (s))
2. B (s, r) = N (r) – N (s)
3. Node r determines F(s, r). (F (s,r) can be chosen from B(s,r) to cover U(s,r)).

Table 3.6 shows the result of BSP-NC-ALGO-3.

Table 3.6: Result of BSP-NC-ALGO-3

s	r	U	B	**F**
φ	6	11,10,8,4,3,1	9,7,5,2	9,2,7
6	7	12	11,8,4	8

6	2	Φ	3,1	[ ]
6	9	Φ	10	[ ]
7	8	Φ	12	[ ]

Result of BSP-NC-ALGO-3 clearly shows that the total numbers of forward nodes are 5 while total number of nodes is 12.

Network coding integrated BSP-NC-ALGO- (1-3):

1. For source node, take out 1-Hop and 2-Hop Neighbors
2. Take out Forwarding Nodes by using BSP-NC-ALGO-(1,2,3) as illustrated earlier
3. Apply network coding concept to using BSP-NC-ALGO-(1, 2, and 3).
   a) For each node, FIFO (First in First Out) queue of packets is created to forward packets.
   b) Keep a track of hash table also.
   c) Probability of every neighbor having packet in output queue is shown by the table.
4. Select each forwarding node from forward list of algo 1, algo 2 and algo 3. If forward probability of each packet is >= 0.5 then perform XOR of all the packets & then broadcast.

3.3. Experiment and Results

We have find out 1-hop & 2-hop neighbor nodes for every node. Random probability of packets at the 1-hop nodes is additionally taken [65].

3.3.1 Results BSP-NC-ALGO-1

Table 3.7 shows the comparison of old BSP subtracting schemes with BSP-NC-ALGO-1.

Table 3.7: Comparison of old BSP subtracting schemes with BSP-NC-ALGO-1

Number of Nodes	Number of Forwarding Nodes - Probabilistic Scheme	Number of Forwarding Nodes - Counter-based Scheme	Number of Forwarding Nodes - Distance-based Scheme	Number of Forwarding Nodes - Location-based Scheme	Number of Forwarding Nodes - Cluster-based Scheme	Number of Forwarding Nodes -BSP-NC-ALGO-1
20	16	15	14	13	12	11
30	24	22	20	20	19	18
40	32	31	29	29	28	27
50	40	37	35	33	32	31
60	48	45	42	42	42	40
70	56	52	48	47	46	45
80	64	58	55	54	53	51
90	72	67	63	62	61	59
100	80	72	68	67	67	66

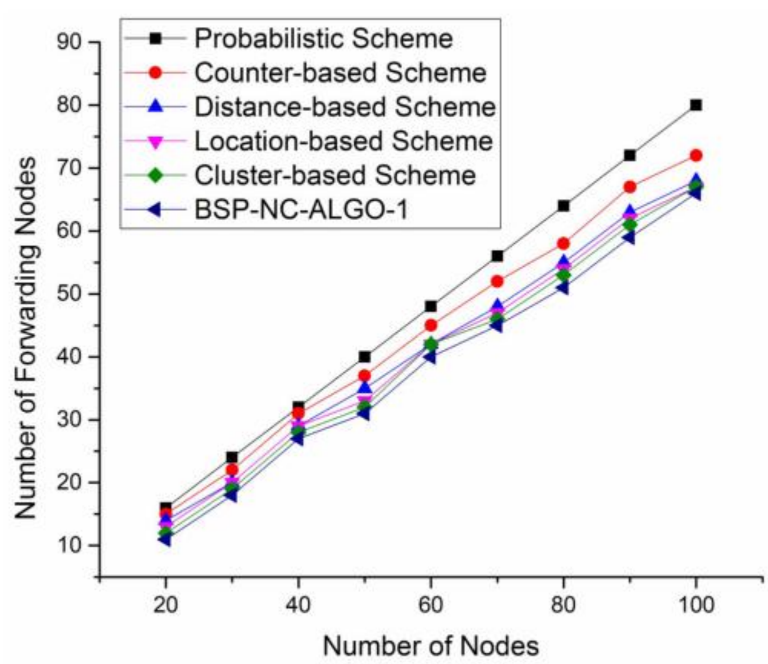

Figure 3.5: Comparison of old BSP subtracting schemes with BSP-NC-ALGO-1

Table 3.8 shows the "number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using" BSP-NC-ALGO-1 keeping transmitter range as 20.

Table 3.8: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-1

Number of Nodes	Number of Forwarding Nodes
20	12
30	20
40	28
50	33
60	41
70	48
80	53
90	60
100	68

Figure 3.6 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping transmitter range as 20.

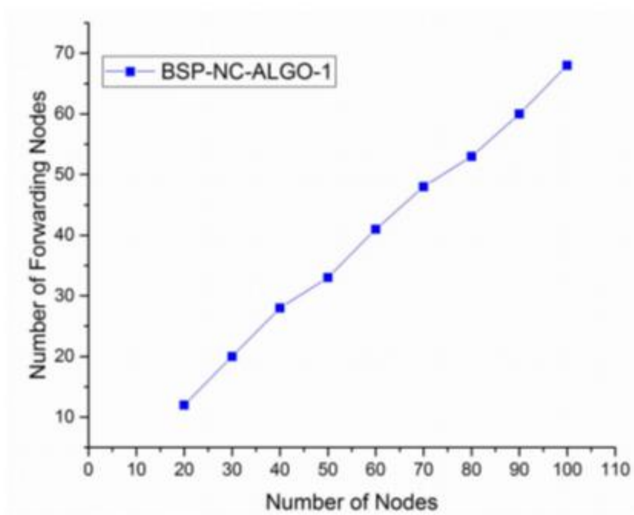

Figure 3.6: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-1

3.3.2 Results BSP-NC-ALGO-2

Table 3.9 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping transmitter range as 20.

Table 3.9: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-2

Number of Nodes	Number of Forwarding Nodes
20	10
30	18
40	21
50	23
60	29
70	31
80	40
90	47
100	49

Figure 3.7 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping transmitter range as 20.

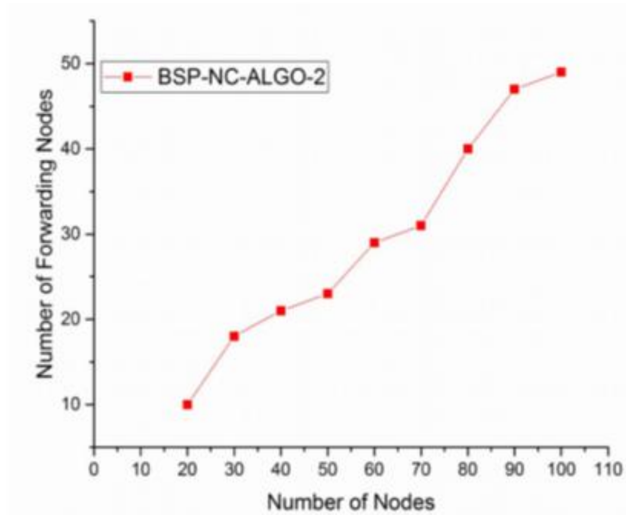

Figure 3.7: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-2

3.3.3 Results BSP-NC-ALGO-3

Table 3.10 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping transmitter range as 20.

Table 3.10: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-3

Number of Nodes	Number of Forwarding Nodes
20	9
30	11
40	18
50	20
60	24
70	30
80	32

90	40
100	41

Figure 3.8 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping transmitter range as 20.

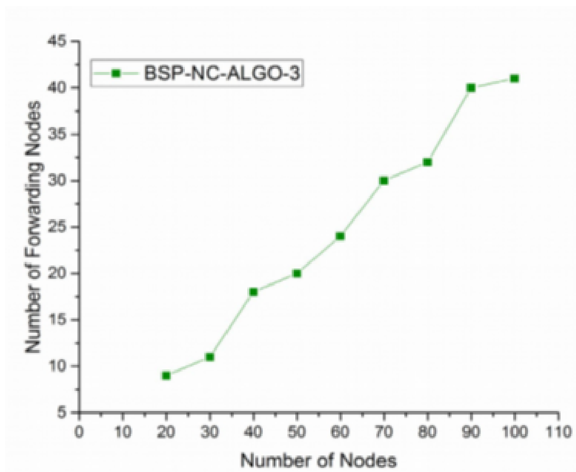

Figure 3.8 Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-3

3.3.4 Comparative Analysis

Table 3.11 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO- (1-3) and keeping transmitter range as 20.

Table 3.11: Comparison of BSP-NC-ALGO-1, 2 & 3 keeping transmitter range as 20

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-3
20	12	10	9
30	20	18	11
40	28	21	18
50	33	23	20
60	41	29	24
70	48	31	30
80	53	40	32
90	60	47	40
100	68	49	41

Figure 3.9 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO- (1-3) and keeping transmitter range as 20.

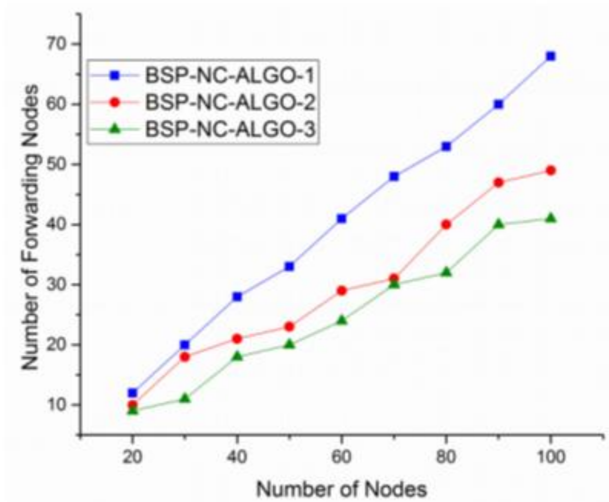

Figure 3.9: Comparison of BSP-NC-ALGO-1, 2 & 3 keeping transmitter range as 20

Table 3.12 illustrates number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping transmitter range as 25.

Table 3.12: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-1

Number of Nodes	Number of Forwarding Nodes
20	11
30	18
40	27
50	31
60	40
70	45
80	51
90	59
100	66

Figure 3.10 shows the "number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using" BSP-NC-ALGO-1 keeping transmitter range as 25.

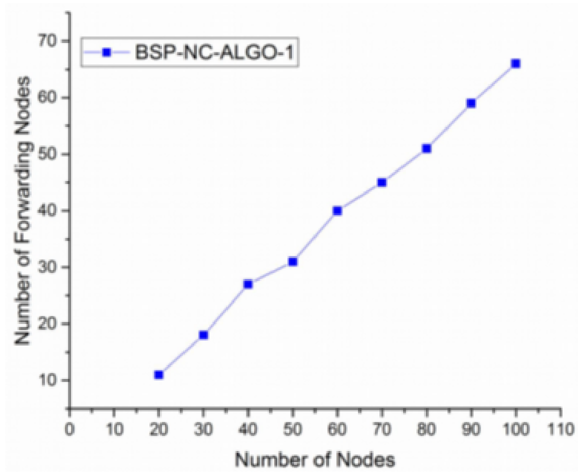

Figure 3.10: Number of Nodes Vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-1

Table 3.13 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping transmitter range as 25.

Table 3.13: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-2

Number of Nodes	Number of Forwarding Nodes
20	8
30	12
40	20
50	21
60	25
70	30
80	40
90	43
100	44

Figure 3.11 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping transmitter range as 25.

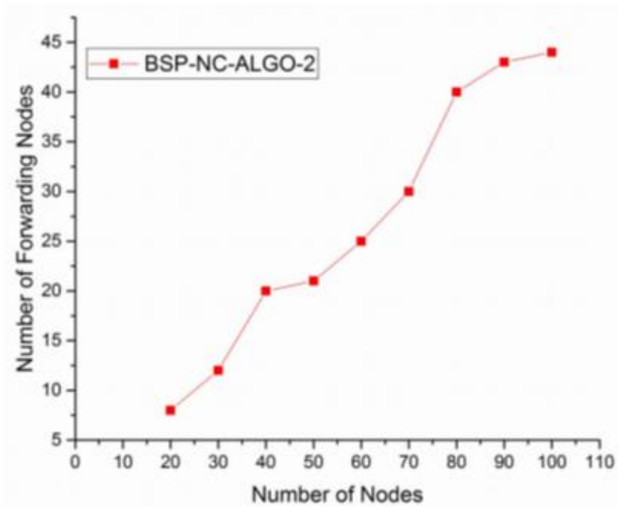

Figure 3.11: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-2

Table 3.14 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping transmitter range as 25.

Table 3.14: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-3

Number of Nodes	Number of Forwarding Nodes
20	8
30	10
40	15
50	18
60	22
70	29
80	31
90	35
100	38

Figure 3.12 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping transmitter range as 25.

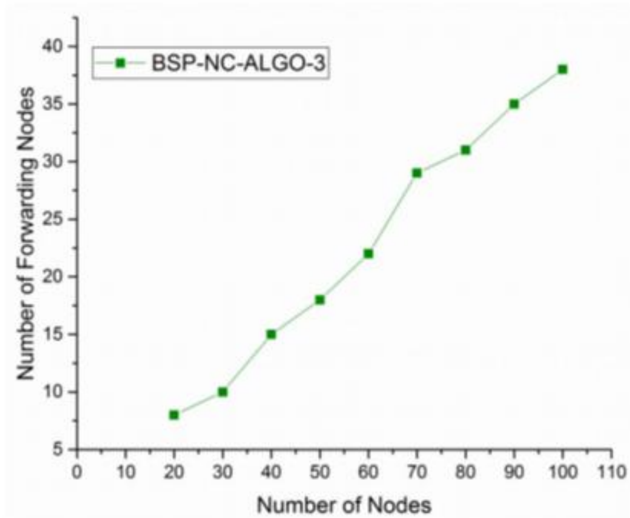

Figure 3.12: Number of Nodes Vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-3

Table 3.15 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) and keeping transmitter range as 25.

Table 3.15: Comparison of BSP-NC-ALGO-1, 2 & 3 based on keeping transmitter range as 25

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3
20	11	8	8
30	18	12	10
40	27	20	15
50	31	21	18
60	40	25	22
70	45	30	29
80	51	40	31
90	59	43	35
100	66	44	38

Figure 3.13 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) and keeping transmitter range as 25.

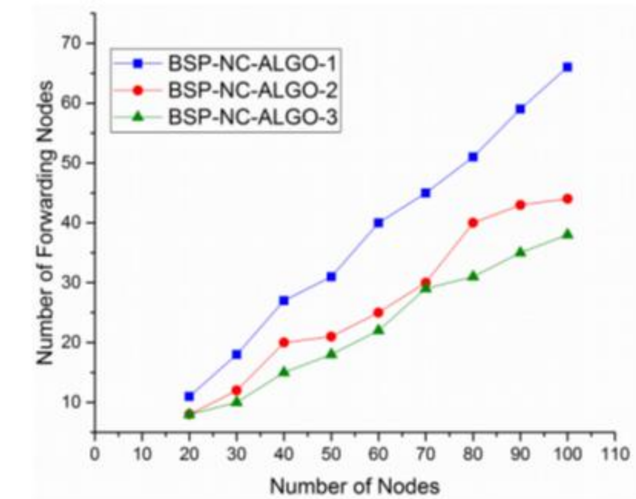

Figure 3.13: Comparison of BSP-NC-ALGO-1, 2 & 3 keeping transmitter range as 25

Result has been tested on the increasing number of total nodes, transmitter range and average node degree. Table 3.16 confirms number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping Average Node degree as 5.

Table 3.16: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-1

Number of Nodes	Number of Forwarding Nodes
20	10
30	19
40	28
50	32
60	40
70	48
80	50
90	58
100	66

Table 3.16 illustrates the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping Average Node degree as 5.

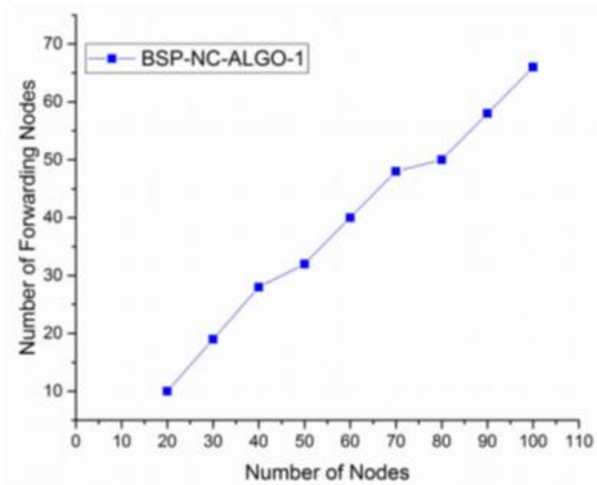

Figure 3.14: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-1

Table 3.17 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping Average Node degree as 5.

Table 3.17: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-2

Number of Nodes	Number of Forwarding Nodes
20	9
30	16
40	20
50	22
60	25
70	31
80	37
90	41
100	45

Figure 3.15 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping Average Node degree as 5.

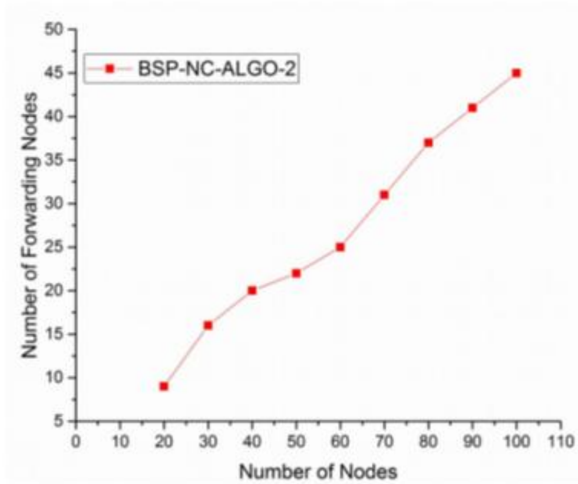

Figure 3.15: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-2

Table 3.18 illustrates the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping Average Node degree as 5.

Table 3.18: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-3

Number of Nodes	Number of Forwarding Nodes
20	8
30	10
40	16
50	20
60	23
70	28
80	32
90	38
100	40

Figure 3.16 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping Average Node degree as 5.

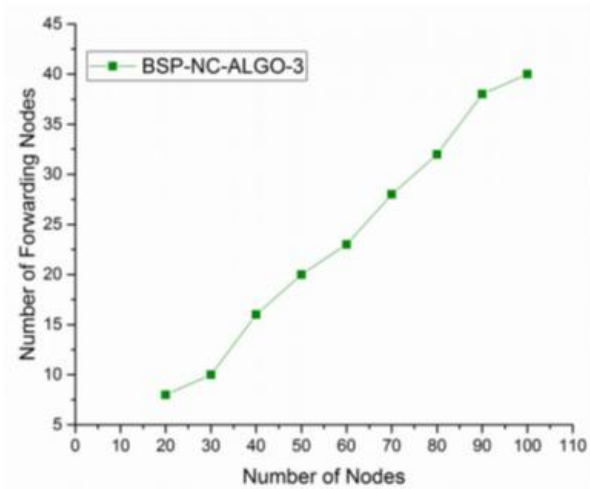

Figure 3.16: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 5 using BSP-NC-ALGO-3

Table 3.19 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) keeping Average Node degree as 5.

Table 3.19: Comparison of BSP-NC-ALGO-1, 2 & 3 based on keeping Average Node degree as 5

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3
20	10	9	8
30	19	16	10
40	28	20	16
50	32	22	20
60	40	25	23
70	48	31	28
80	50	37	32
90	58	41	38
100	66	45	40

Figure 3.17 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) keeping Average Node degree as 5.

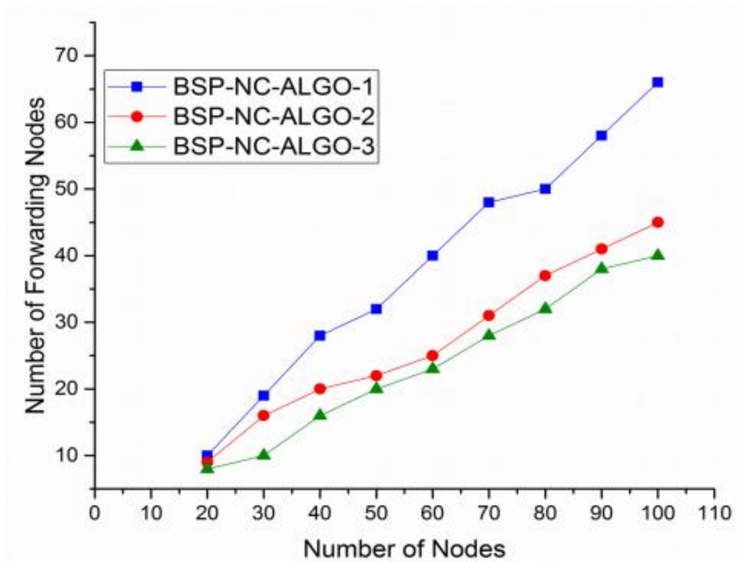

Figure 3.17: Comparison of BSP-NC-ALGO-1, 2 & 3 based on keeping Average Node degree as 5

Table 3.20 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping Average Node degree as 10.

Table 3.20: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-1

Number of Nodes	Number of Forwarding Nodes
20	9
30	16
40	25
50	30
60	38
70	43
80	49
90	55
100	61

Figure 3.18 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-1 keeping Average Node degree as 10.

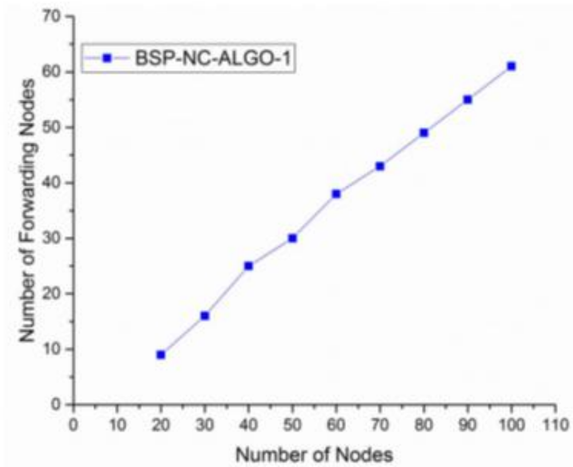

Figure 3.18: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-1

Table 3.21 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping Average Node degree as 10.

Table 3.21: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-2

Number of Nodes	Number of Forwarding Nodes
20	8
30	14
40	18
50	20
60	23
70	29
80	34
90	40
100	41

Figure 3.19 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-2 keeping Average Node degree as 10.

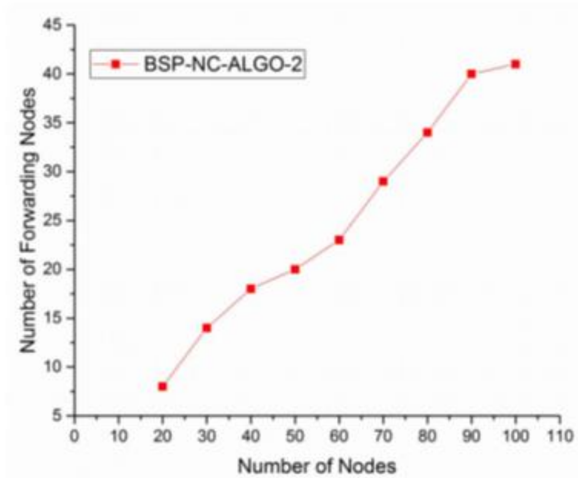

Figure 3.19: Number of Nodes Vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-2

Table 3.22 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping Average Node degree as 10.

Table 3.22: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-3

Number of Nodes	Number of Forwarding Nodes
20	6
30	10
40	13
50	18
60	21
70	25
80	30
90	32
100	38

Figure 3.20 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-3 keeping Average Node degree as 10.

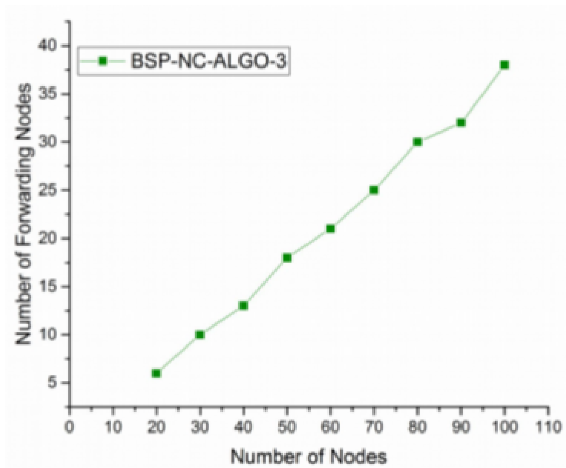

Figure 3.20: Number of Nodes vs. Number of Forwarding Nodes keeping Average Node degree as 10 using BSP-NC-ALGO-3

Table 3.23 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) keeping Average Node degree as 10.

Table 3.23: Comparison of BSP-NC-ALGO-1, 2 & 3 keeping Average Node degree as 10

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3
20	9	8	6
30	16	14	10
40	25	18	13
50	30	20	18
60	38	23	21
70	43	29	25
80	49	34	30
90	55	40	32
100	61	41	38

Figure 3.21 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-3) keeping Average Node degree as 10.

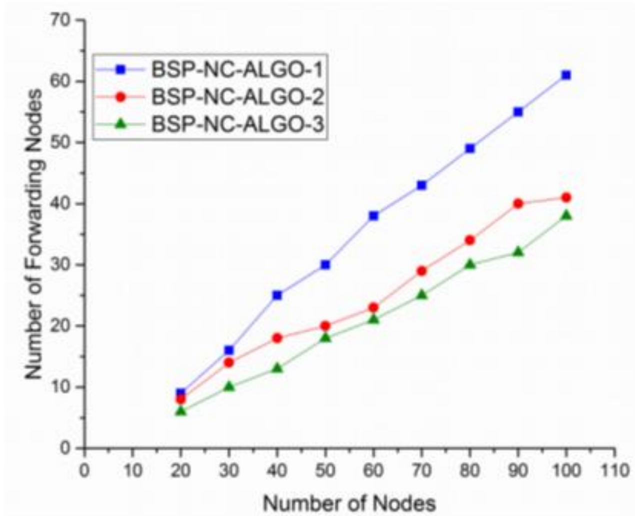

Figure 3.21: Comparison of BSP-NC-ALGO-1, 2 & 3 keeping Average Node degree as 10

The simulation shown within the Figure 3.21 clearly states that BSP-NC-ALGO- 3 requires minimum number of nodes for network coding in comparison other to BSP-NC-ALGO 1 & 2.

3.3.5 Results BSP-NC-ALGO-4

Here, we have proposed an algorithm which will consider incoming traffic degree and outgoing traffic degree of a node along with sum of input and output bandwidth of a node decide forwarding nodes.

In this Algorithm BSP-NC-ALGO-4,

$D_{in}(i)$ represents incoming traffic degree of network node

$D_{out}(i)$ represents outgoing traffic degree of network node

$Conn_{ji}$, $conn_{ij} = 1$ if connection is present ,else = 0

$S_{in}(i)$ & $S_{out}(i)$ represents the dominance, sum of input and output link bandwidth (bw)

"$N_{in}(i)$ " represents the incoming nucleus of a node,

"$N_{out}(i)$ " represents the outgoing nucleus of a node,

"N(i) " represents the nucleus of a node, "P" represents the probability of a node of packet transmission.

ALGORITHM:

BSP-NC-ALGO-4: For (all nodes i =1 to M), calculate from step 1-7

1. $D_{in}(i) = {}_{j=1}\Sigma^{M} conn_{ji}$
2. $D_{out}(i) = {}_{j=1}\Sigma^{M} conn_{ij}$
3. $S_{in}(i) = {}_{j=1}\Sigma^{M} bw_{ji}$
4. $S_{out}(i) = {}_{j=1}\Sigma^{M} bw_{ij}$
5. $N_{in}(i) = D_{in}(i) * (S_{in}(i) / D_{in}(i))^{p}$
6. $N_{out}(i) = D_{out}(i) * (S_{out}(i) / D_{out}(i))^{p}$
7. $N(i) = N_{out}(i) * (N_{in}(i) / N_{out}(i))^{p}$
8. Choose that node "i" for which the value of nucleus N(i) is *maximum* and apply network coding.

Table 3.24 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-4 keeping transmitter range as 20.

Table 3.24: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-4

Number of Nodes	Number of Forwarding Nodes
20	8
30	9
40	16
50	18
60	21
70	27
80	29
90	36
100	37

Figure 3.22 shows the number of forwarding nodes needed for Network coding in contrast to actual number of nodes present in the system using BSP-NC-ALGO-4 keeping transmitter range as 20.

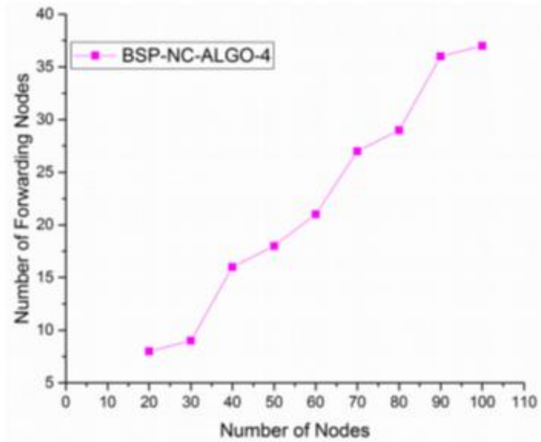

Figure 3.22: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-4

Table 3.25 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-4) and keeping transmitter range as 20.

Table 3.25: Comparison of BSP-NC-ALGO-1, 2, 3 & 4 keeping transmitter range as 20

Num ber of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-4
20	12	10	9	8
30	20	18	11	9
40	28	21	18	16
50	33	23	20	18
60	41	29	24	21
70	48	31	30	27
80	53	40	32	29
90	60	47	40	36
100	68	49	41	37

Figure 3.23 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-4) and keeping transmitter range as 20.

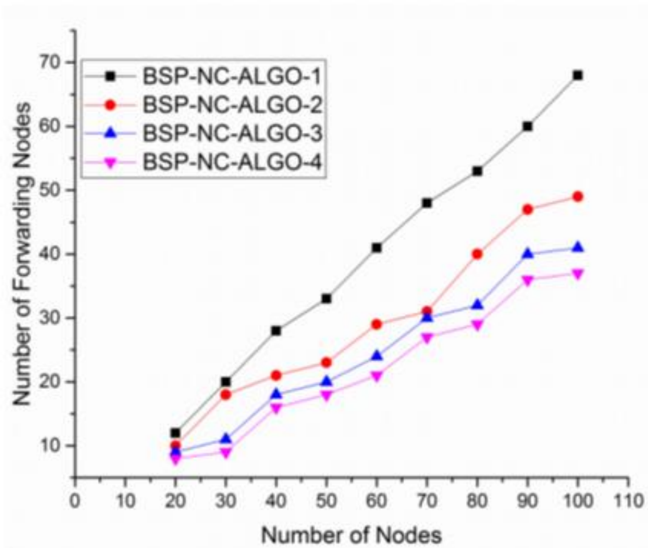

Figure 3.23: Comparison of BSP-NC-ALGO-1, 2, 3 & 4 keeping transmitter range as 20

Table 3.26 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-4 keeping transmitter range as 25.

Table 3.26: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-4

Number of Nodes	Number of Forwarding Nodes
20	7
30	8
40	13
50	15
60	19
70	25

80	26
90	30
100	32

Figure 3.24 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-4 keeping transmitter range as 25.

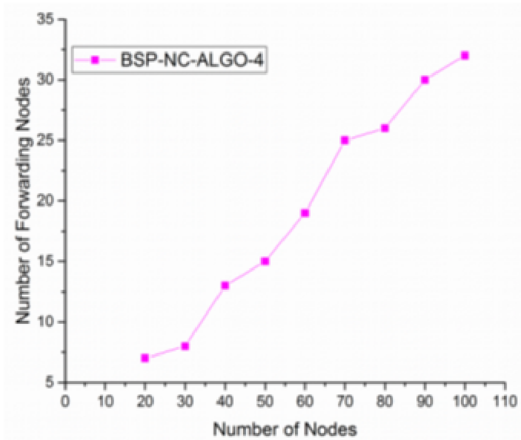

Figure 3.24: Number of Nodes vs. Number of Forwarding Nodes keeping transmitter range as 25 using BSP-NC-ALGO-4

Table 3.27 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-4) and keeping transmitter range as 25.

Table 3.27: Comparison of BSP-NC-ALGO-1, 2, 3 & 4 keeping transmitter range as 25

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-4
20	11	8	8	7
30	18	12	10	8
40	27	20	15	13
50	31	21	18	15
60	40	25	22	19
70	45	30	29	25
80	51	40	31	26

90	59	43	35	30
100	66	44	38	32

Figure 3.25 shows the comparison for number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-4) and keeping transmitter range as 25.

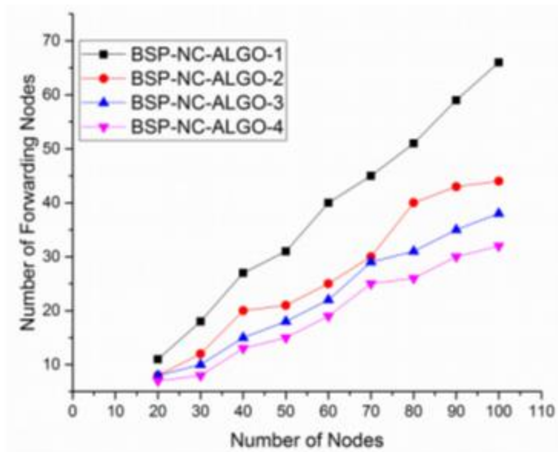

Figure 3.25: Comparison of BSP-NC-ALGO-1, 2, 3 & 4 keeping transmitter range as 25

3.4. Summary and Discussion

Broadcasting in a MANET has generally extraordinary attributes from that in different networks. It could bring about mindful redundancy, collision and contention. Network coding may affect the outline of the design of new networking and information dissemination protocols. The simulation clearly demonstrates that BSP-NC-ALGO-3 requires minimum number of nodes when integrated along with the concept of network coding in contrast with other two algorithms (BSP-NC-ALGO-1 & 2) to reduce number of forwarding nodes for reducing the effect of broadcast storm problem.

BSP-NC-ALGO -4 works on the concept of incoming traffic degree, outgoing traffic degree of network node, and sum of input and output link bandwidth of a node of packet transmission. BSP-NC-ALGO -4 requires minimum number of nodes in contrast to BSP-NC-ALGO 1, 2 and 3 & thus performs better.

CHAPTER 4

ADDRESS BIT VECTOR & NETWORK TOMOGRAPHY INTEGRATED NETWORK CODING SCHEME IN WIRELESS NETWORK

4.1 Introduction

Network coding (NC) [68] is a strategy which is used to join a set of packets together into one solitary packet. Packet grouping is typically carried out in 2 routes: (i) By playing out a straight mix of the packets (ii) By playing out XOR operation. Advantages for NC can be in unwavering quality, storage & security. Enhanced unwavering quality can be unmistakably shown in Figure 4.1 which represents NC (network coding) utilizing XOR operator. S_2 & S_1 are 2 sources & 4 collectors D_4, D_3, D_2 and D_1. By noticing packet P_2 as packet generated by S_2 & P_1 created by S_1 & recipients need to get all packets, a middle hub, R can XOR the two packets to make network-coded packet and send it to collectors which are able to remove packet P_1 having P_2 & skilled to extricate packet P_2 having packet P_1.

Network coding implementation is difficult due to its complex nature and the trouble of mixing the packets with existing framework. Including a consistent combination of packets, for example, XOR requires each hub in the system to have the ability to create such mixtures. Late work recommends that Network coding turns out to be excessively complex when managing Gaussian elimination [60].

Reliability can likewise be accomplished utilizing linear combinations rather than the XOR operation; nonetheless it is all the more expensive as far as intricacy. For the most part hubs have restricted computational capabilities which ruin them from performing complicated arithmetic operations.

A Network coding scheme called Network coding codes with origin information to remove redundancy of combination is introduced in this paper. Ordinarily, in those applications, sink hub starts round of gathering where hubs communicate processed information to the sink node. In systems where the sink is found far from the hubs and with dynamic paths, a gathering based convention, for example, CTP [60] can be productively used to makes the courses. Principle reason with utilizing network coding for accomplishing unwavering quality is that coding countless will prompt an expansive enough overhead that defeats its points of interest.

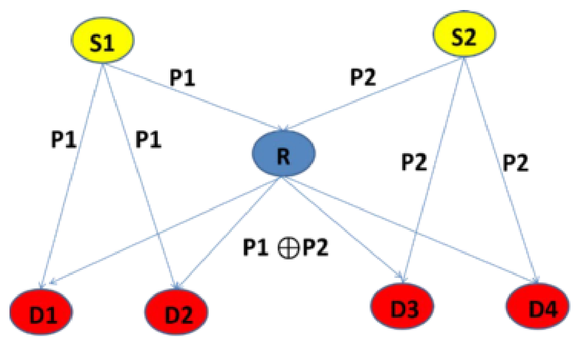

Figure 4.1: Network Coding using XOR Operator

4.2 System Model

Our network coded algorithm is an NC convention which accomplishes reliability for information gathering application for remote systems. We accept a two-level system with static courses where every hub conveys its packet to a relay node. Thus, this transfer will convey got packets to the sink. We accept systems where transfers, as a rule, are sufficiently far that they don't catch each other. This will include complexity while accomplishing reliability since caught packets by transfers are major in recouping packets.. In our algorithm NC is performed utilizing

XOR operation. Just, when data packet is obtained, hubs will XOR the substance with at least one packet which is accessible in its cradle. Then, resulting packet is sent to the next hop.

To recognize the packets which were XORed during NC, address of every packet is incorporated into the header of the coded message. Be that as it may, this represents a vast overhead on the span of the coded packets which at times can surpass the measure of payload. To conquer this subject, our algorithm uses bit addresses where it alludes to the identity of the actual originating packets.

For an illustration as appeared in Figure. 4.2, consider that relay node got packet P_3 and packet P_4 and needs to network-code them together. Accept for effortlessness, there are just eight hubs in system with a bit vector of size as appeared in Figure. 4.2.

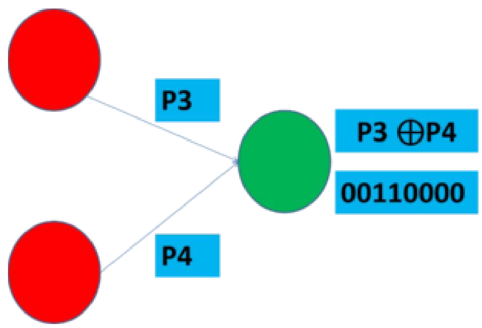

Figure 4.2: Binary coding of addresses

When NC (Network coding) is performed using XOR operation, Set relating bits that allude to the address of originating packets to bit value as '1'. For example shown in figure 4.2, bit 3 & 4 will be set to show that the coded packets contain data from hub 3 & hub 4. a bigger piece vector will be utilized in bigger systems.

4.3 Algorithm Design

Principle thought behind our algorithm is to infuse enough excess to accomplish unwavering quality without exhausting the power hold of the remote hubs. Our algorithm design indicates 3 actions for hubs, each with a different assignment. In Figure 4.3, hubs at the primary level will send recorded information and can code packets. Relay nodes are in charge of decoding, coding & sending packets. Sink hub is in charge of uncoding packets.

Gathering begins when the sink starts a round for information accumulation. At the point when hubs get the demand from the sink, hubs will start information exchange by sending the uncoded parcels, for instance, Node 2 will send bundle P_2 to the relay hub. Then, it sits tight for a pre-decided time to catch neighboring parcels. For every new uncoded bundle got, the hubs will XOR its substance with its content with its list of uncoded parcels each one in turn. Just un-coded packets are coded, as demonstrated in figure 4.3 where hubs don't action decoding. For instance, if hub 2 catches parcels P_1 & P_3, it will along these lines send, to the relay hub, $P_2 \oplus P_1$ and afterward $P_3 \oplus P_1 \oplus P_2$. Additionally, for every parcel, this will set corresponding bit addresses in headers. Normally, in the primary packet the hub will set bit location 2 & 1, & in the second bundle it will set-piece area 3, 2, & 1.

4.4 Algorithm1: Network Coding at Nodes

Pseudo code: proposed algorithm

A. if !Network_Coding() then
B. Hold_up for sink packet()
C. else
D. Begin Network_Coding()
E. Timer.begin()
F. Bit_Address = 1<<a
G. P_a.Network_coded=0
H. send_Packet(P_a)
I. end if
J. if Hear_Packet(P_b) then
K. if Heard_Packet_Is_Uncoded(P_b) then
L. $P_a = P_a$ XOR P_b
M. P_a.Coded=1
N. Address = 1<<a | 1<<b
O. Forward(P_a)
P. end if
Q. if Timer.fired() then

R. end Timer()

S. end if

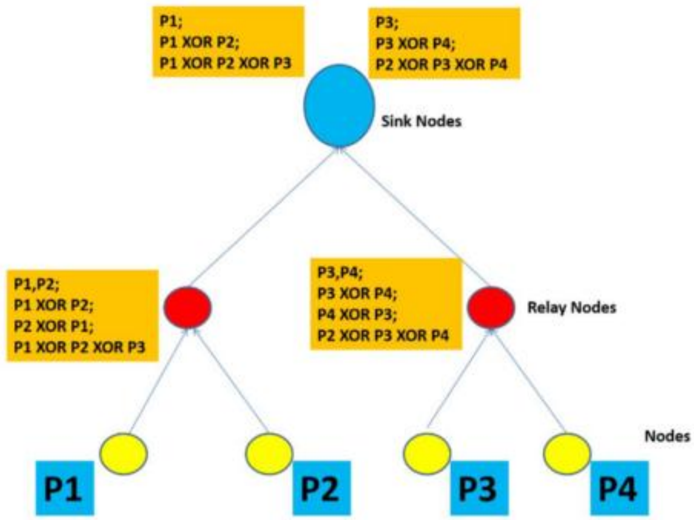

Figure 4.3: Wireless network

After gathering of packets (single or combined) at the relay hubs, parcels are ordered whether they are uncoded or coded. The relay hub will interpret these bundles in planning for sending as appeared in Algorithm 2. The point of the interpreting calculation is to achieve all parcels in their un-coded unique format. Decode calculation functions as follows: Initial step of translating begins with un-coded bundles. It will emphasize over uncoded bundles each in turn. Calculation checks if the un-coded parcels exist in coded bundles.

4.5. Algorithm 2: Decoding at Relay Nodes

Pseudo code: proposed algorithm

A. Hold_up for Start();

B. if Heard_Packet(P_a) then

C. if P_a.uncoded == 1 then

D. Uncoded_Buffer.add(P_a)

E. else

F. Coded_Buffer.add(P_a)

G. end if

H. end if

I. while(All_Packets_Uncoded() | m<n)

J. for a=1:size.Uncoded_Buffer

K. for b=1:size.Coded_Buffer

L. P = Codded_Bufferb)>

M. if Cardinality(P_a) >1 then

N. if Uncoded_Buffer(a).exists then

O. P = P XOR Uncoded_Buffer(a)

P. P.Address = P.Address XOR Uncoded_Buffer(a). Address

Q. if Sum_up(P) ==1 then

R. Uncoded_Buffer.add(P)

S. end if

T. end if

U. end if

V. end for

W. end for

X. end while

Sum_up() test is finished by summing up bits in the bit address vector. If outcome is 1, at that point we have achieved the first uncoded bundle. This algorithm finds these packets by summing up bits in bit address vector. If message is still un-decodable now, it will be put in a different cradle anticipating further packets if any is gotten a short time later. When task of decoding completes, the relay will get ready packets to be sent.

First: P_1

Second: P_1 XOR P_2

Third: P_1 XOR P_2 XOR P_3

Fourth: P_4 XOR P_3 XOR P_2 XOR P_1

At last the sink will get the parcels from all transfers and will play out the interpreting calculation depicted before.

4.6. Experiment and Result

To confirm the accuracy of the proposed work, we have tested the above said algorithm on three distinct systems. The systems comprise of a solitary sink, 5 relay nodes and we shifted the quantity of hubs from 10 to 20 as indicated Figure. 4.4, 4.5 & 4.6.

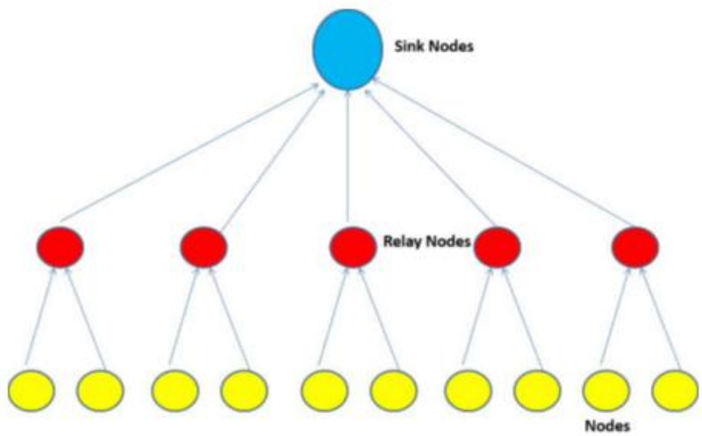

Figure 4.4: Network with Ten nodes

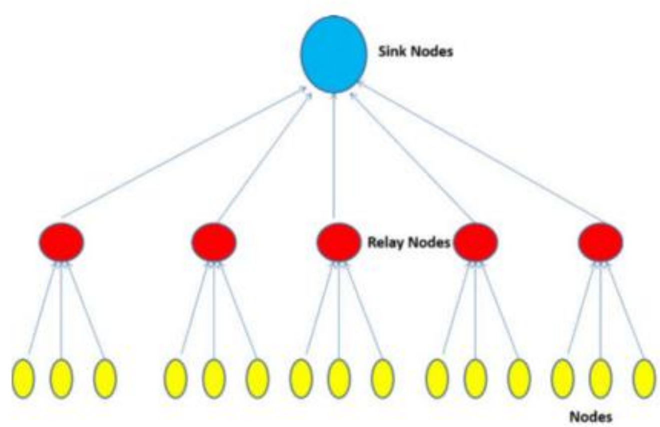

Figure 4.5: Network with Fifteen nodes

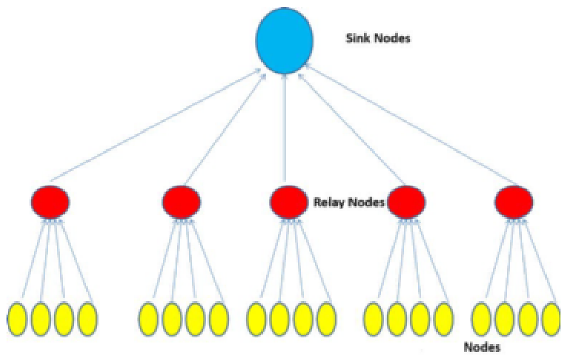

Figure 4.6: Network with Twenty nodes

Simulation for each of the 3 systems was rehashed ten times. For every emphasis, we diverged amount of packet loss from zero percent to almost hundred percent of the first parcels sent by hubs. The packet loss was randomized. As appeared in Figure 4.7, 4.8 & 4.9, our algorithm could recoup most parcels even in unforgiving conditions where the error presented was hundred percent. The achievement proportion was up to 90% percent when the system comprises of ten hubs as appeared in Figure 4.7.

Table 4.1 illustrates the simulation end results for “Number of packet lost vs. success delivery ratio” in a network with ten nodes using BSP-NC-ALGO-5.

Table 4.1: Simulation Results- Network with Ten nodes using BSP-NC-ALGO - 5

Number of Packet Lost	Success Delivery Ratio
0	100
1	99
2	98
3	97
4	96
5	95
6	94
7	93
8	92
9	91
10	90

Figure 4.7 shows the simulation end results for "Number of packet lost vs. success delivery ratio" in a network with ten nodes using BSP-NC-ALGO-5.

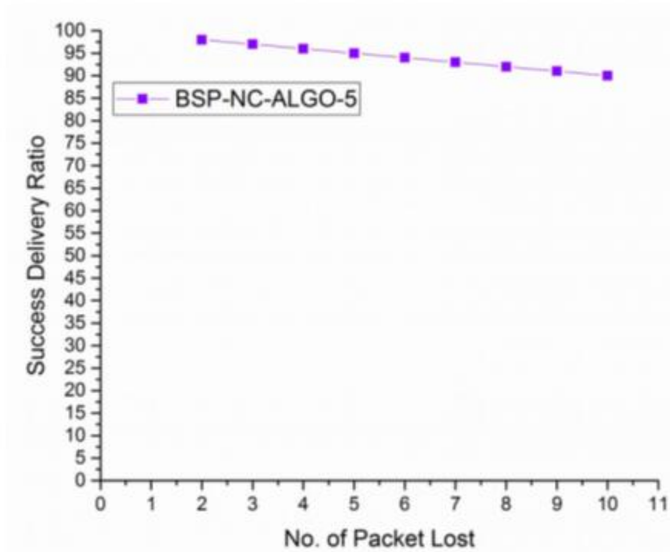

Figure 4.7: Simulation Results- Network with Ten nodes

Table 4.2 represents the simulation end results for "Number of packet lost vs. success delivery ratio" in a network with Fifteen nodes using BSP-NC-ALGO-5.

Table 4.2: Simulation Results- Network with Fifteen nodes using BSP-NC-ALGO-5

Number of Packet Lost	Success Delivery Ratio
0	100
1	99
2	98
3	97
4	96
5	95
6	92
7	90
8	89
9	88
10	86

Figure 4.8 mentions the simulation end results for "Number of packet lost vs. success delivery ratio" in a network with fifteen nodes using BSP-NC-ALGO-5.

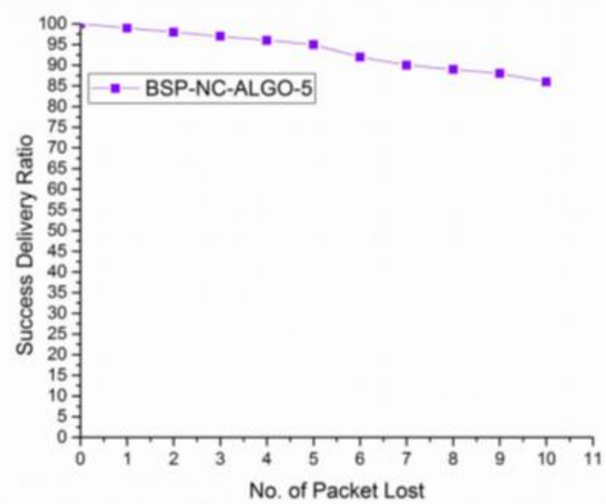

Figure 4.8: Simulation Results- Network with Fifteen nodes

In Figure 4.8 and 4.9, the execution weakens in light of the fact that the quantity of parcels sent is less since the quantity of hubs per relay node increased. In this manner, when the quantity of hubs increment in a similar hop, the quantity of overheard parcels is lessened & thusly the quantity of sent packets. Our algorithm just sends the coded & original un-coded packet from nearby neighbors.

To exhibit the impact of including more hubs, we reenacted a system of 8 hubs and 4 transfers. Then, we quantified the quantity of packet sent & the level of redundancy. The redundancy is figured as for original parcels sent. For a case, an excess of two hundred percent implies that the quantity of bundles sent is double the quantity of unique parcels.

Table 4.3 represents the simulation end results for "Number of packet lost vs. success delivery ratio" in a network with twenty nodes using BSP-NC-ALGO-5.

Table 4.3: Simulation Results- Network with Twenty nodes using BSP-NC-ALOGO-5

Number of Packet Lost	BSP-NC-ALGO-5
0	100
1	99
2	98
3	97
4	96.5
5	94
6	90
7	88
8	85
9	83
10	80

Figure 4.9 represents the simulation end results for "Number of packet lost vs. success delivery ratio" in a network with twenty nodes using BSP-NC-ALGO-5.

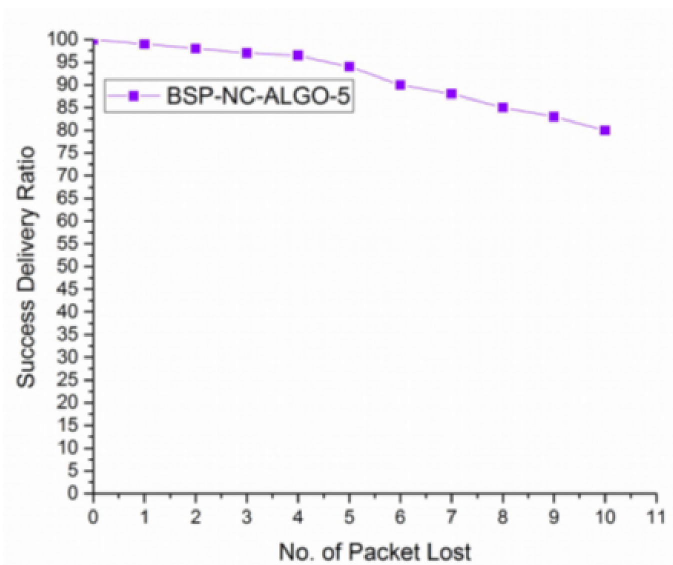

Figure 4.9: Simulation Results- Network with Twenty nodes

Table 4.4 reveals the simulation end results for "Number of packet sent vs. total number of nodes" using BSP-NC-ALGO-5.

Table 4.4: Total number of packets sent with respect to number of nodes using BSP-NC-ALGO-5

Total Number of Nodes	Number of Packets Sent
10	15
15	23
20	30
25	35
30	42

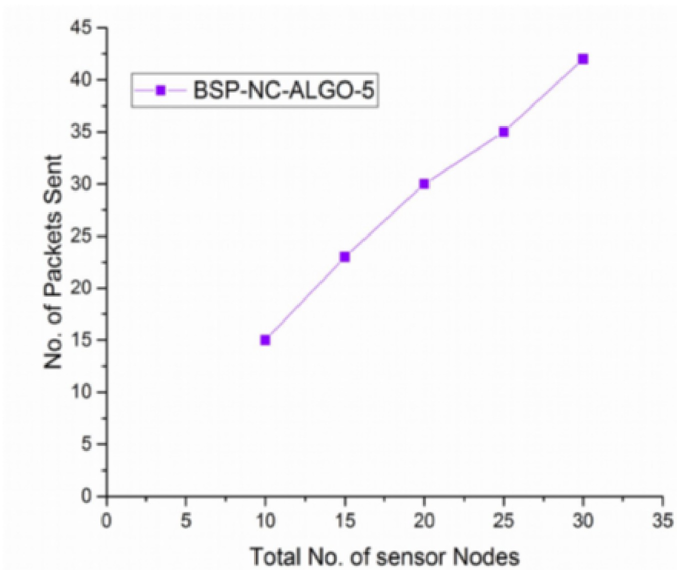

Figure 4.10: Total number of packets sent with respect to number of nodes

Considering Figures 4.10 and 4.11, quantity of hubs is set to twenty while keeping network with neighboring hops to one, the repetition is 1.4. If 100% of the bundles are lost; the framework won't have the capacity to recoup all parcels. In this way, in unforgiving conditions, it is accepted that the number of hubs ought to be increased.

Table 4.5 conveys the simulation results for % of packet sent vs. total number of nodes using BSP-NC-ALGO-5.

Table 4.5: "% of packets sent with respect to number of nodes" using BSP-NC-ALGO-5

Total Number of Nodes	% of Packets Sent
10	182
15	152
20	127
25	118
30	105

Figure 4.11 shows the simulation results for % of packet sent vs. total number of nodes using BSP-NC-ALGO-5.

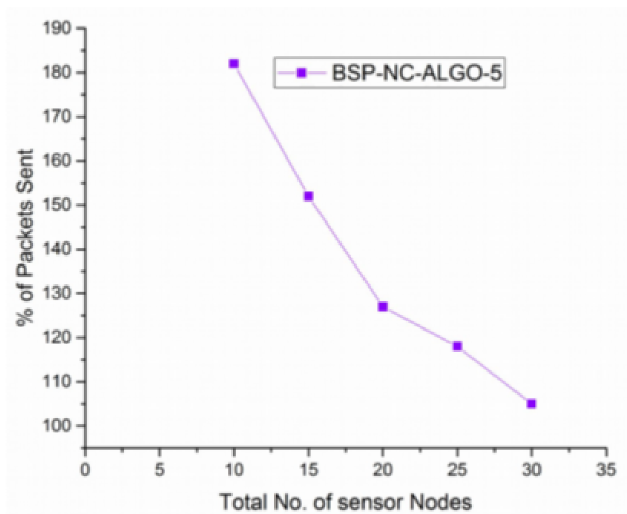

Figure 4.11: % of packets sent with respect to number of nodes

Table 4.6 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-5 while keeping transmitter range as 20.

Table 4.6: Number of Nodes V/s. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-5

Number of Nodes	Number of Forwarding Nodes
20	7
30	9
40	14

50	16
60	20
70	27
80	28
90	33
100	35

Figure 4.12 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-5 while keeping transmitter range as 20.

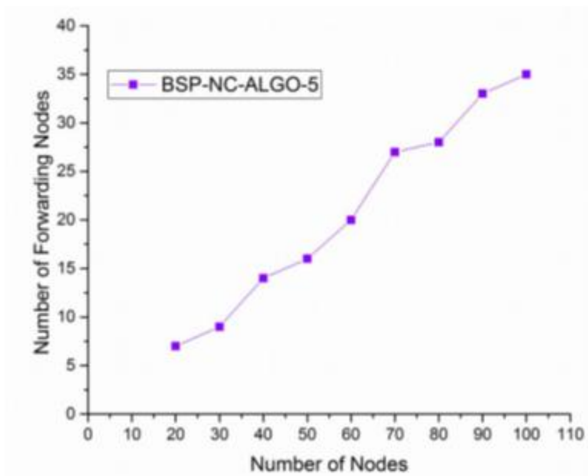

Figure 4.12: Number of Nodes V/s. Number of Forwarding Nodes keeping transmitter range as 20 using BSP-NC-ALGO-5

Table 4.7 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-5).

Table 4.7: Comparison of BSP-NC-ALGO-1, 2, 3, 4 & 5

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-4	Number of Forwarding Nodes using BSP-NC-ALGO-5

20	11	8	8	7	7
30	18	12	10	8	9
40	27	20	15	13	14
50	31	21	18	15	16
60	40	25	22	19	20
70	45	30	29	25	27
80	51	40	31	26	28
90	59	43	35	30	33
100	66	44	38	32	35

Figure 4.13 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-5).

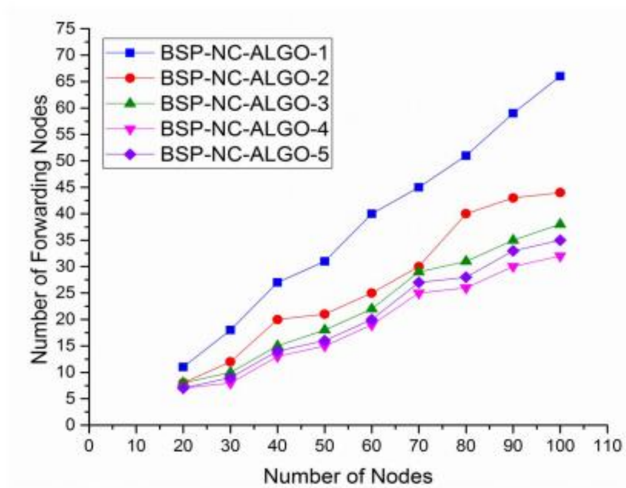

Figure 4.13: Comparative analysis BSP-NC-ALGO-(1-5)

Approach 2

Consider Figure. 4.14 with 7 leaves (1,2,3,4,10,11,12) and 5 intermediate nodes (5,6,7,8,9). Presume that nodes 1, 10 act as sources S_1, S_2 and send probes p_1 = [10], p_2 = [01] in that order. Rest leaves act as receivers.

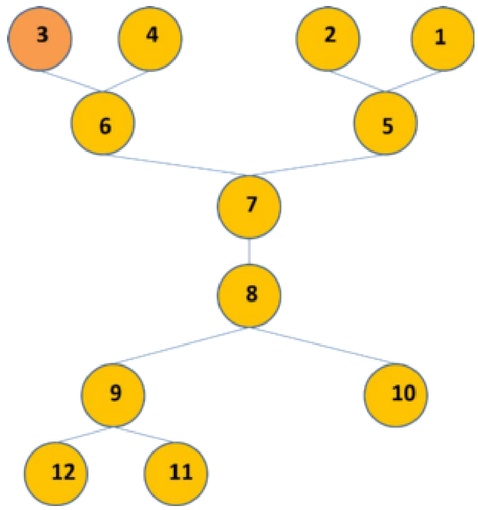

Figure 4.14: Network with 2 Source, 7 leaf and 5 Intermediate Nodes

Algorithm:

Iteration 1: Consider the set SET which comprises of all leaf nodes.

1. Two leaf nodes start working as sources S_1, S_2; sending probes P_1, P_2 respectively.

All other leaves S - $\{S_1; S_2\}$ act as receivers.

2. Partition SET (into SET_1 U SET_2 U SET_3) as follows.

   a) SET_1 contains S_1 and all receivers that receive p_1.

   b) SET_2 contains S_2 and all receivers that receive p_2.

   c) SET_3 contains all receivers that receive $p_3 = p1 \oplus p2$.

3. If $SET_3 \neq \phi$, replace the original graph with the three sets SET_1, SET_2, SET_3.

4. If $SET_3 = \phi$,, replace the original graph with two sets SET_1, SET_2, connected through a single edge.

Iteration i:

1. Consider any 1 previously identified SET_i and repeat.

2. As done earlier, two leaves in SET_i act as sources S_1 and S_2 and all remaining nodes in SET_i act as receivers.

3. Node A_i that connects SET_i to the network will act as an aggregate receiver; whatever packet is received by A_i will be multicasted and received by all leaves in SET that are not in SET_{i}.

4. Repeat the exact same procedure as in iteration 1 to reveal the structure of SET_i . Connect the component to the network depending on what packet is received by A_i.

5. Remove vertices of degree two this way and so on.

Illustration of Algorithm

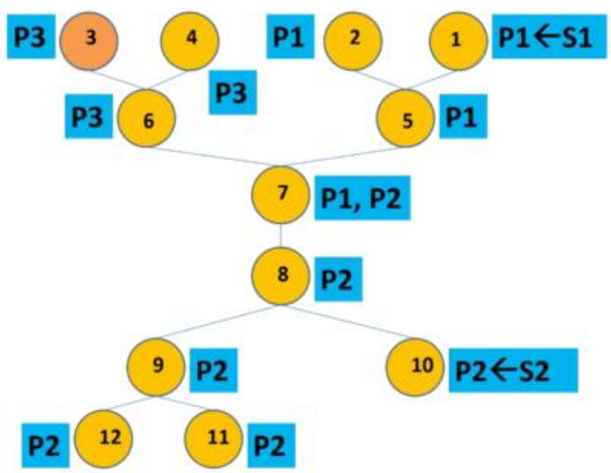

Figure 4.15: Illustration of Algorithm

In figure 4.15, Intermediate node 5 receives p_1 & forwards it to leaf 2 and to node 7. Likewise, node 8 receives p_2 & forwards it to node 9 (which in turn forwards it to leaves 11, 12) and to node 7. Probe packets p_1 & p_2 turn up at node 7, which adds them, creates packet $p_3 = p_1 \oplus p_2 =$ [11], & aheads p_3 to node 6, which in turn aheads it to leaves 3, 4. Leaf 2 obtains p_1, leaves 11, 12 obtain p_2 & leaves 3, 4 obtain $p_3 = p_1 \oplus p_2$.

Partition:

SET_1 = {1, 2}; leaf nodes having packet p_1 and source S_1

SET_2 = {10, 11,12} ; leaf nodes having packet p_2 and source S_2

SET_3 = {3, 4} ; leaf nodes having packet p_3

Table 4.8 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping value of Probe messages (M=1) and iteration “ i” as 1.

Table 4.8: “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6; M=1; i=1

““Percentage Loss Probability ‘p’”on Every Link”	“Percentage Wrong Inference of topology SET”
0.5	5
1	9
1.5	14
2	19
2.5	24
3	27
3.5	30
4	33
4.5	38
5	41
5.5	45
6	47
6.5	49
7	51
7.5	55
8	57
8.5	59
9	60
9.5	62
10	64

Figure 4.16 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping different values of Probe messages (M=1) and iteration “ i” as 1.

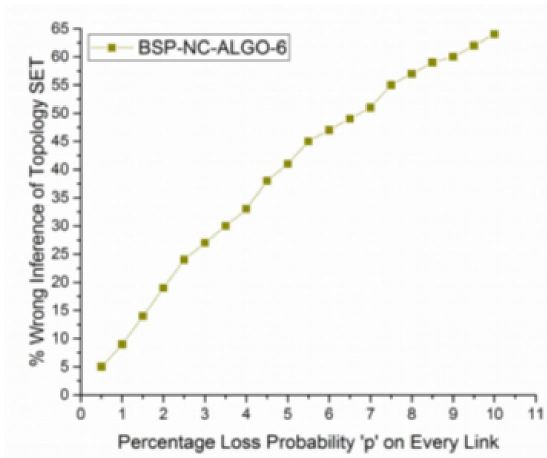

Figure 4.16: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=1; i=1

Table 4.9 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping values of Probe messages (M=2) and iteration " i" as 1.

Table 4.9: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=2; i=1

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET"
0.5	0.5
1	0.8
1.5	0.9
2	1.2
2.5	1.3
3	4.6
3.5	5.5
4	7
4.5	9
5	10
5.5	11
6	15
6.5	16

7	18
7.5	20
8	22
8.5	24
9	26
9.5	27
10	30

Figure 4.17 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=2) and iteration " i" as 1.

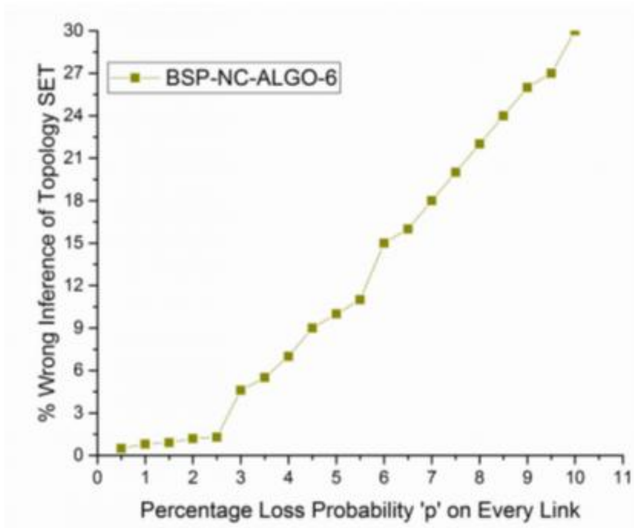

Figure 4.17: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=2; i=1

Table 4.10 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=4) and iteration " i" as 1.

Table 4.10: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=4; i=1

““Percentage Loss Probability ‘p’”on Every Link”	“Percentage Wrong Inference of topology SET”
0.5	0.5
1	0.8
1.5	0.9
2	1.2
2.5	1.1
3	1.7
3.5	2.5
4	3.6
4.5	4.3
5	5.6
5.5	6.3
6	7.1
6.5	7.6
7	8.3
7.5	8.7
8	9.3
8.5	9.7
9	10.3
9.5	11.1
10	12

Figure 4.18 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping different values of Probe messages (M=4) and iteration “ i” as 1.

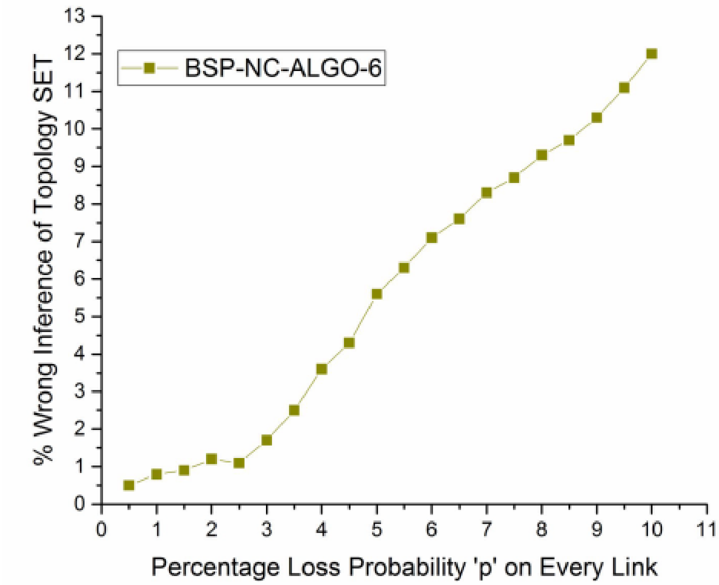

FIGURE 4.18: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=4; i=1

Table 4.11 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=8) and iteration " i" as 1.

Table 4.11: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=8; i=1

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET"
0.5	0.1
1	0.12
1.5	0.13
2	0.14
2.5	0.16
3	0.2
3.5	0.24
4	0.3

4.5	0.34
5	0.39
5.5	0.44
6	0.47
6.5	0.54
7	0.59
7.5	0.64
8	0.67
8.5	0.69
9	0.73
9.5	0.75
10	0.77

Figure 4.19 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping different values of Probe messages (M=8) and iteration “ i” as 1.

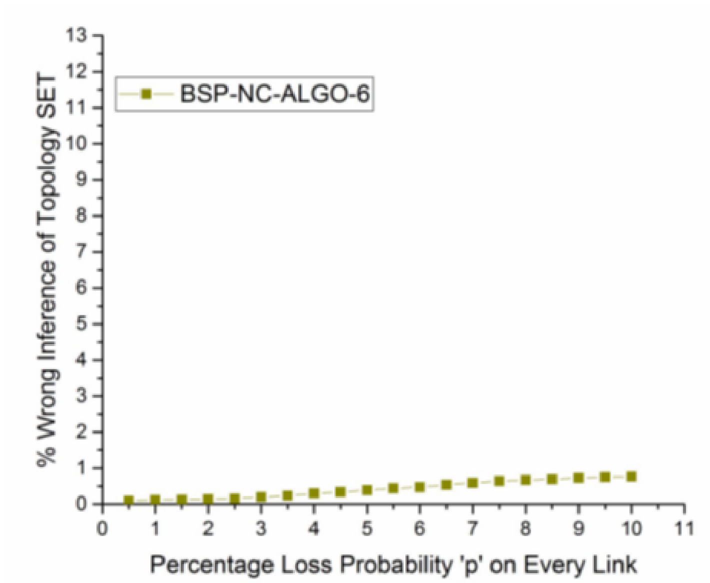

FIGURE 4.19: “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6; M=8; i=1

Table 4.12 shows the comparative analysis of “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping diverse values of messages (Probe M=1 , 2 , 4 and 8) and iteration “ i” as 1.

Table 4.12: “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6; i=1; M=1,2,4 & 8

““Percentage Loss Probability ‘p’”on Every Link”	“Percentage Wrong Inference of topology SET” ;M=1	“Percentage Wrong Inference of topology SET” ;M=2	“Percentage Wrong Inference of topology SET” ; M=4	“Percentage Wrong Inference of topology SET” ;M=8
0.5	5	0.5	0.5	0.1
1	9	0.8	0.8	0.12
1.5	14	0.9	0.9	0.13
2	19	1.2	1.2	0.14
2.5	24	1.3	1.1	0.16
3	27	4.6	1.7	0.2
3.5	30	5.5	2.5	0.24
4	33	7	3.6	0.3
4.5	38	9	4.3	0.34
5	41	10	5.6	0.39
5.5	45	11	6.3	0.44
6	47	15	7.1	0.47
6.5	49	16	7.6	0.54
7	51	18	8.3	0.59
7.5	55	20	8.7	0.64
8	57	22	9.3	0.67
8.5	59	24	9.7	0.69
9	60	26	10.3	0.73
9.5	62	27	11.1	0.75
10	64	30	12	0.77

Figure 4.20 shows the comparative analysis of “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping diverse values of messages (Probe M=1 , 2 , 4 and 8) and iteration “ i” as 1.

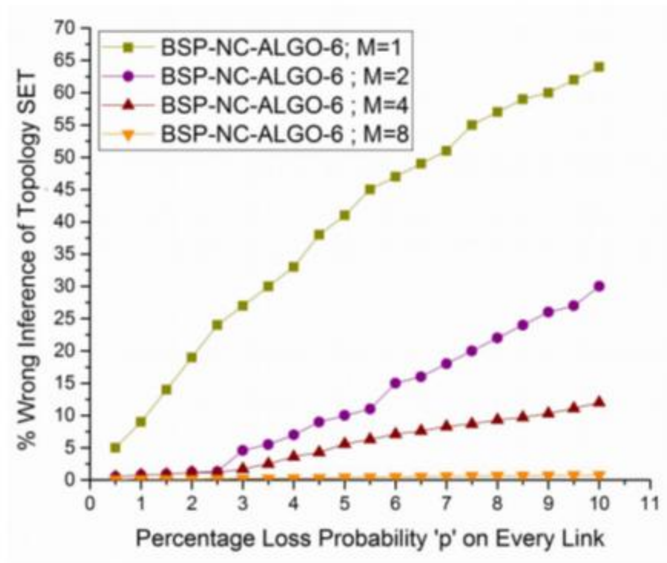

Figure 4.20: Comparative analysis of BSP-NC-ALGO for i=1 & M=1, 2, 4 & 8

Table 4.13 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=1) and iteration " i" as 2.

Table 4.13: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=1; i=2

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET"
0.5	2.5
1	4.5
1.5	7
2	9.5
2.5	12
3	13.5
3.5	15
4	16.5
4.5	19
5	20.5

5.5	22.5
6	23.5
6.5	24.5
7	25.5
7.5	27.5
8	28.5
8.5	29.5
9	30
9.5	31
10	32

Figure 4.21 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=1) and iteration " i" as 2.

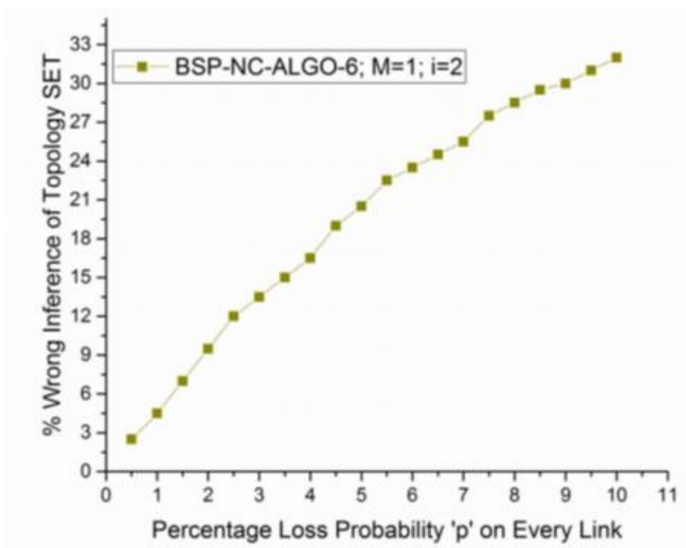

Figure 4.21: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=1; i=2

Table 4.14 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=2) and iteration " i" as 2.

Table 4.14: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=2; i=2

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET"
0.5	0.25
1	0.4
1.5	0.45
2	0.6
2.5	0.65
3	2.3
3.5	2.75
4	3.5
4.5	4.5
5	5
5.5	5.5
6	7.5
6.5	8
7	9
7.5	10
8	11
8.5	12
9	13
9.5	13.5
10	15

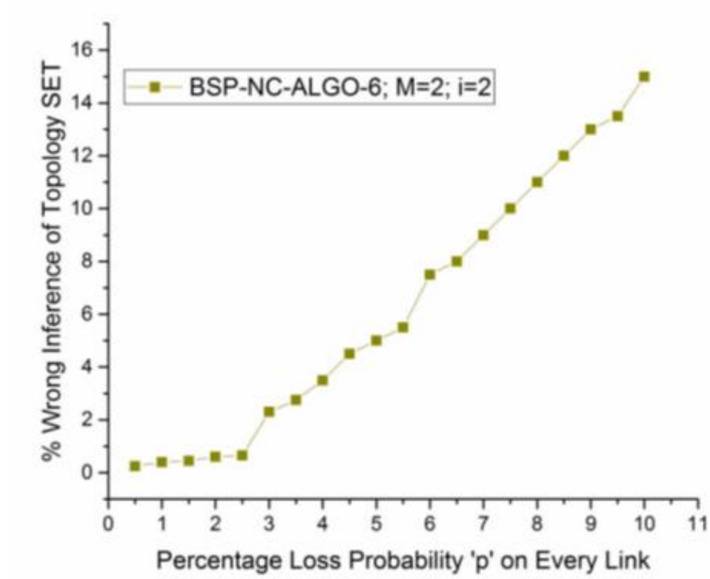

Figure 4.22: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=2; i=2

Table 4.15 shows the "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping different values of Probe messages (M=4) and iteration " i" as 2.

Table 4.15: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=4; i=2

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET"
0.5	0.25
1	0.4
1.5	0.45
2	0.6
2.5	0.55
3	0.85
3.5	1.25
4	1.8
4.5	2.15
5	2.8

5.5	3.15
6	3.55
6.5	3.8
7	4.15
7.5	4.35
8	4.65
8.5	4.85
9	5.15
9.5	5.55
10	6

Figure 4.23 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping different values of Probe messages (M=4) and iteration “ i” as 2.

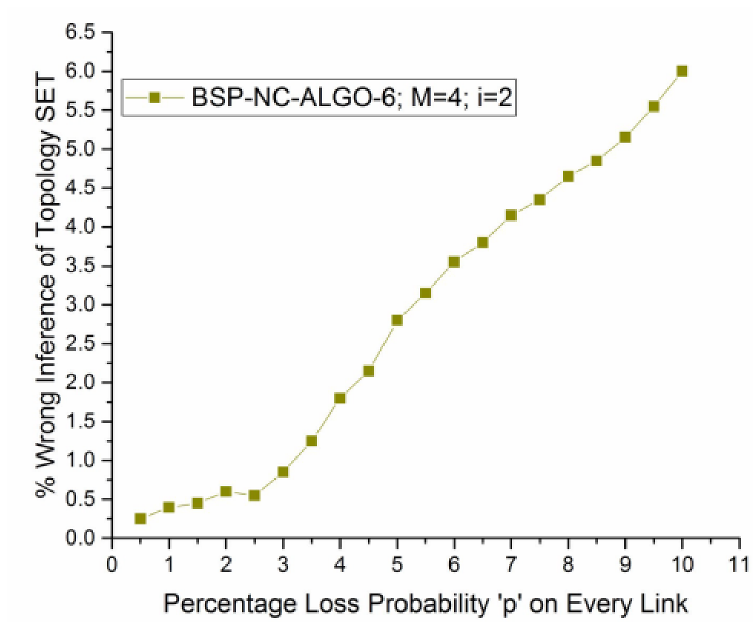

Figure 4.23: “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6; M=4; i=2

Table 4.16: “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6; M=8; i=2

““Percentage Loss Probability ‘p’”on Every Link”	“Percentage Wrong Inference of topology SET”
0.5	0.08
1	0.09
1.5	0.10
2	0.11
2.5	0.12
3	0.15
3.5	0.18
4	0.23
4.5	0.26
5	0.30
5.5	0.34
6	0.36
6.5	0.42
7	0.45
7.5	0.49
8	0.52
8.5	0.53
9	0.56
9.5	0.58
10	0.59

Figure 4.24 shows the “Percentage Loss Probability ‘p’” Vs. “Percentage Wrong Inference of topology SET” using BSP-NC-ALGO-6 keeping different values of Probe messages (M=8) and iteration “ i” as 2.

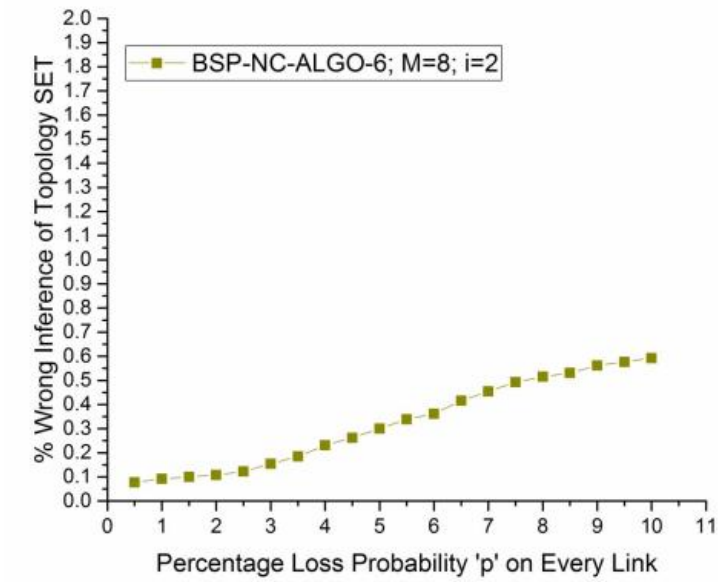

Figure 4.24: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=8; i=2

Table 4.17 shows the comparative analysis of "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping diverse values of messages (Probe M=1 , 2 , 4 and 8) and iteration " i" as 2.

Table 4.17: "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6; M=1, 2, 4 & 8; i=2

""Percentage Loss Probability 'p'"on Every Link"	"Percentage Wrong Inference of topology SET" ;M=1	"Percentage Wrong Inference of topology SET" ;M=2	"Percentage Wrong Inference of topology SET" ; M=4	"Percentage Wrong Inference of topology SET" ;M=8
0.5	2.5	0.25	0.25	0.08
1	4.5	0.4	0.4	0.09
1.5	7	0.45	0.45	0.10
2	9.5	0.6	0.6	0.11
2.5	12	0.65	0.55	0.12
3	13.5	2.3	0.85	0.15
3.5	15	2.75	1.25	0.18

4	16.5	3.5	1.8	0.23
4.5	19	4.5	2.15	0.26
5	20.5	5	2.8	0.30
5.5	22.5	5.5	3.15	0.34
6	23.5	7.5	3.55	0.36
6.5	24.5	8	3.8	0.42
7	25.5	9	4.15	0.45
7.5	27.5	10	4.35	0.49
8	28.5	11	4.65	0.52
8.5	29.5	12	4.85	0.53
9	30	13	5.15	0.56
9.5	31	13.5	5.55	0.58
10	32	15	6	0.59

Figure 4.25 shows the comparative analysis of "Percentage Loss Probability 'p'" Vs. "Percentage Wrong Inference of topology SET" using BSP-NC-ALGO-6 keeping diverse values of messages (Probe M=1 , 2 , 4 and 8) and iteration " i" as 2.

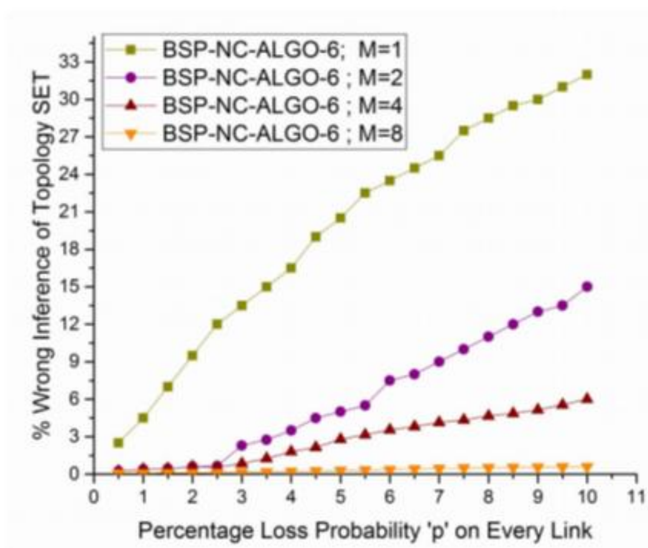

Figure 4.25: Comparative analysis of BSP-NC-ALGO for i=2 & M=1, 2, 4 & 8

Table 4.18 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-6).

Table 4.18: Comparison of BSP-NC-ALGO-1, 2,3,4,5 & 6

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-4	Number of Forwarding Nodes using BSP-NC-ALGO-5	Number of Forwarding Nodes using BSP-NC-ALGO-6
20	11	8	8	7	7	6
30	18	12	10	8	9	7
40	27	20	15	13	14	13
50	31	21	18	15	16	14
60	40	25	22	19	20	18
70	45	30	29	25	27	22
80	51	40	31	26	28	24
90	59	43	35	30	33	27
100	66	44	38	32	35	29

Figure 4.26 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-6).

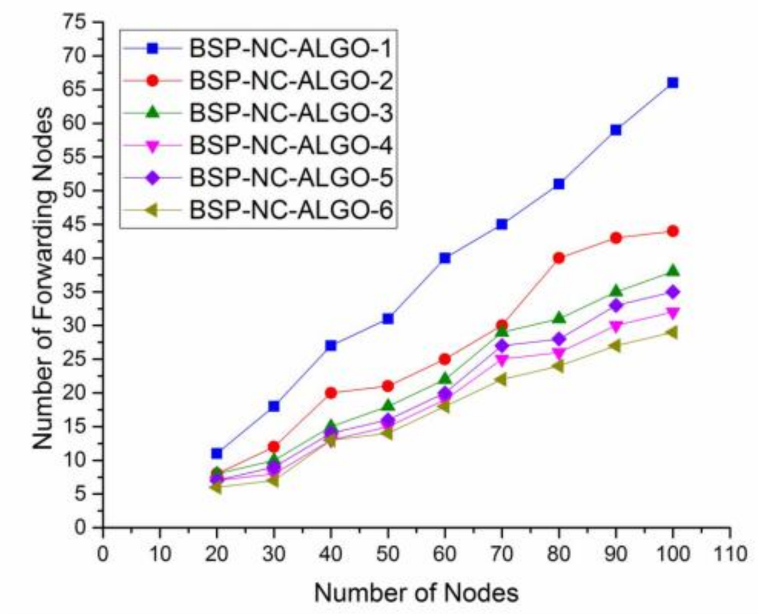

Figure 4.26: Comparison of BSP-NC-ALGO-(1-6)

4.7. Summary and Discussion

The proposed algorithm BSP-NC-ALGO-5 uses the XOR operator to network-code the different packets. Besides, this utilizes bit addresses to which brings down the measure of bytes sent per packet. Also, utilizing bit addresses to help in recognizing the quantity of packets incorporated into each coded packet and permits to effortlessly distinguishing which packet are incorporated into the coded message. Testbeds were led on three distinct systems where the quantity of nodes is expanded. "SET Partition theory-based algorithm (BSP-NC-ALGO-6) when integrated with Network Coding" approach distinguishes and comprises nodes based on the fact of bundles of different network coded reduces the Number of Forwarding nodes for re-transmission in contrast to actual number of nodes. Proposed algorithm BSP-NC-ALGO-6 performs better than BSP-NC-ALGO-5.

CHAPTER 5

IMPROVING THE AVERAGE NUMBER OF NETWORK CODED TRANSMISSION IN AODV ROUTING PROTOCOL WITH NETWORK CODING SCHEME

5.1 Introduction

Network Coding is a newly emerged standard to competently broadcast the data in wireless scenarios, where message flows originating from diverse sources are joined to improve the throughput & raise robustness. As opposed to long-established store & forward style [78], this actualizes a store and forward procedure, where each node stocks approaching packets in particular buffer & forward their XORed output.

Primarily, an individual network administrator has to make out a sub-graph provided that link capacity can uphold the multicast connections. Secondly, the network code can be built self-governing of that subgraph. Then over again, in the dissimilar unicast situations, the matter of choosing a sub-graph & the network code formation must be tackled jointly. The determination of the sub-graph is typically displayed as a breakthrough issue on flows in the network, while the network code formation is algebraic issue. Since we need to embark upon both issues equally, we bound to basic network codes [79].

The advantage of NC (Network coding) in case of multicast network can be elucidated by the network presented in Figure 5.1 (a), where source node is "A", & nodes "D" & "E"are the two beneficiaries. The transmission links in the broadcast system have the limit one. Source node A has two (2) bits, "b_2" and "b_1" to forward to beneficiaries. In the commencement, we capture the customary multicast scheme w/o network coding approach has appeared in Figure 5.1(b). Bit "b_1" can arrive "D" along the way "A" to "B" & then "B" to "D". Bit "b_2" can arrive "E" along the way "A" to "C" and then "C" to "E". When node "F" gets both bits, it requires forwarding them to node "G" in sequence. Assume it advances bit b_1 first, at that point "D" and "E" gets both bits "b_2" and "b_1". At the point when node "F" gets the both bits, it primarily makes an XOR operation to them. At that point it forwards the XORed bit to node "G". When beneficiaries "D" or "E" get XORed version, it can convalesce first bit "b_1" &"b_2" by XORing the XORed bit with another [80] .

An essential end result that started the eagerness for network coding is that it can offer the throughput gain. When the 'N' beneficiaries plum the network assets, each beneficiary can get the greatest rate it might like to get, in spite of of the fact that it was utilizing all system assets autonomous from anyone else. Along these lines, network coding can preferably offer accessible network assets [80].

From the perspective of network security, network coding offers both advantages and disadvantages. As a case we again take the butterfly network of Figure. 5.1 (b), if a hacker acquires the coded packet $b1 \oplus b2$, it is unrealistic for him to acquire either b1 or b2. This is a security advantage.

Another point of interest of network coding in this appreciation is that it encourages the utilization of a sub-graph containing numerous ways to every sink node. At the point when network coding is done crosswise over multiple paths, it offers valuable potential outcomes for data security against foes.

Then again network coding at transitional nodes offers some new dangers in the system. Coded packets that include some wrong packet result in more mistaken mix packets.

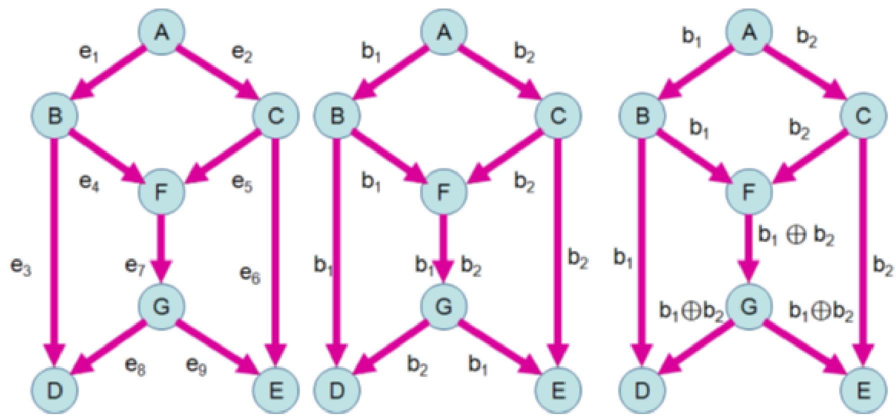

Figure 5.1: Multicast network (a) Butterfly network (b) Multicast scenario without NC (c) Multicast scenario along with NC

5.2 Proposed Algorithm

In our approach, AODV protocol is integrated with Network Coding (NC) approach to make maximum utilization of both the theories. AODV has following main procedures.

Path Discovery

Process begins when source node "S" looks to trade several packets with other node for which it has nil routing information [81]. Every node sustains 2 different counters: a broadcast id & node sequence number.

Following fields are accompanied by a Route Request:

('source-address'; 'source-sequence-number'; 'broadcast-id'; 'destination-address'; dest-sequence-number', 'hop-count')

When an origin node "S" issues a new Route Request, the broadcast__id is increased. When a node obtains a Route Request, and if it has received a Route Request with the same *id* & source address in advance, This node drops the copy of Route Request [81].

Reverse Path Setup

A node preserves proof of the address of the neighbor from which it has attained the first copy of Route Request to set up the reverse path. To go over the network, those reverse path route logs are kept for sufficient time for the Route Request [82].

Forward Path Setup

At the point when the Route Request comes at a node that has a way to beneficiaries, it figures out if the root is current by looking at the destination sequence number. If sequence number in RREQ is higher than that maintained by the moderate node, the moderate node must not utilize its recorded path.

Route Table Management

For every entry, active neighbor"s address is documented through which packets for the given destination are obtained. A neighbor is understood as if it forwards at least 1 packet for that destination.

In the advised algorithm, using AODV from a known graph G, paths between each pair of nodes are determined. Then for every node, the count of reiteration of each of its nexthops for dissimilar destinations are recorded. Next hops are chosen for which the count is maximum. For these next hops of a node, the edges connecting them with the node are retained as they were in the original graph, while others are deleted. Thus, a subgraph of the original graph G" is created. Network coding (NC) algorithm is performed over that sub-graph.

Algorithm (G)

A. By using AODV algorithm, routes are found out for every pair of nodes.

B. **For** ∀ node "p" in graph G, **do**

C. **for** Node "q"=1 to 'n' **do**

D. **for** Node "r"=1 to 'n' **do**

E. m.nexthops ← next_hop for source=q, destination=r

F. Counts ← reiterations for each Nexthop

G. **end for**

H. **end for**

I. p.Max = max(p.Counts), nexthop for which count = max is recorded and stored in p.Neighbors

J. **end for**

K. **For** Nodes p=1 to n **do**

L. **for** q=1 to n **do**

M. **if** q ∈ i.neighbors **then**

N. matrix(p,q)=1

O. **else**

P. matrix(p,q)=0

Q. **endif**

R. **end for**

S. **end for**

T. Graph G is drawn between nodes i & j if matrix(p,q)=1.

U. Network Coding algorithm is implemented on graph G ,sub-graph of G.

5.3. Experiment and Results

Maximum allowed degree of a node is taken as 5 for the end simulation intention and algorithm is tested on 50 number of nodes. Table 5.1 shows the "Average Transmission per Node vs. Forwarding Factor" using traditional network coding (NC) scheme.

Table 5.1: "Average Transmission per Node vs. Forwarding Factor" using Traditional Network Coding Scheme

Forwarding factor	Average Transmission per Node
0.05	2.5
0.1	5
0.15	7.5
0.2	10
0.25	12.5
0.3	15
0.35	17.5
0.4	20
0.45	22.5
0.5	25
0.55	27.5
0.6	30
0.65	32.5
0.7	35
0.75	37.5
0.8	40
0.85	42.5
0.9	45
0.95	47.5
1	50

Figure 5.2 reveals the graph between "Average Number of Transmissions per Node vs. Forwarding Factor" in case old network coding scheme.

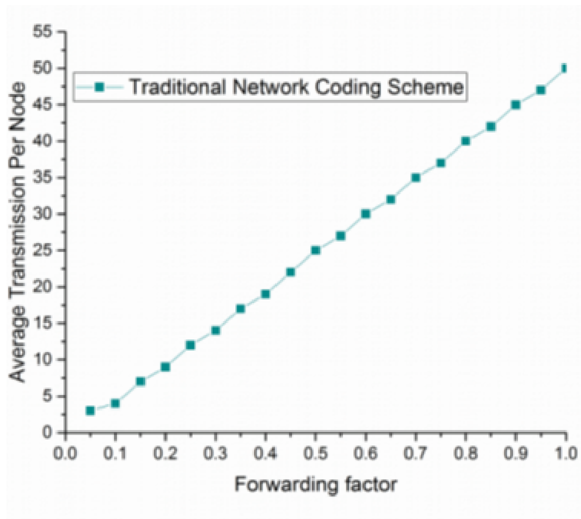

Figure 5.2: "Average Number of Transmissions per Node vs. Forwarding Factor" using Traditional Network Coding Scheme

Table 5.2 shows the "Average Transmission per Node vs. Forwarding Factor" BSP-NC-ALGO-7.

Table 5.2: "Average Number of Transmissions per Node vs. Forwarding Factor" using BSP-NC-ALGO-7

Forwarding factor	Average Transmission Per Node
0.05	1
0.1	1.2
0.15	1.4
0.2	1.5
0.25	1.6
0.3	3.5
0.35	3
0.4	7
0.45	10
0.5	11
0.55	13
0.6	15
0.65	17
0.7	20

0.75	21
0.8	22
0.85	24
0.9	27
0.95	29
1	30

Figure 5.3 reveals graph between "Average Number of Transmissions per Node vs. Forwarding Factor" in our proposed scheme BSP-NC-ALGO-7.

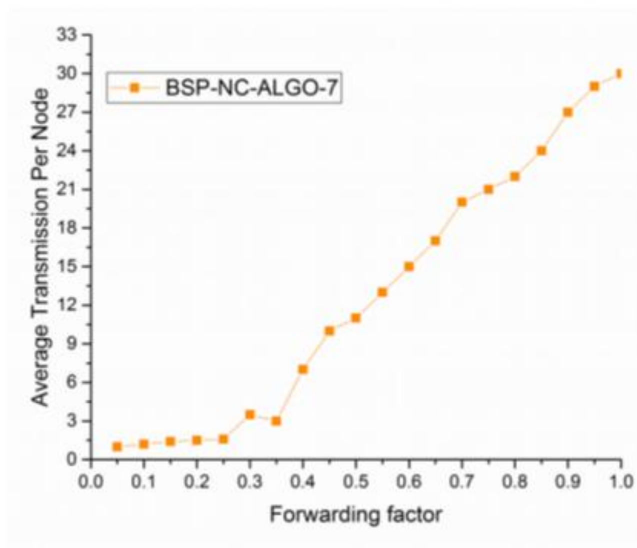

Figure 5.3: "Average Number of Transmissions per Node vs. Forwarding Factor" using BSP-NC-ALGO-7

Table 5.3 shows the Comparison of Traditional Network Coding Scheme and BSP-NC-ALGO-7 based on Average Number of Transmissions per Node versus Forwarding Factor .

Table 5.3: Comparison of Traditional Network Coding Scheme and BSP-NC-ALGO-7 based on "Average Number of Transmissions per Node vs. Forwarding Factor"

Forwarding factor	Average Transmission Per Node using Traditional Network Coding Scheme	Average Transmission Per Node using BSP-NC-ALGO-7
0.05	2.5	1
0.1	5	1.2
0.15	7.5	1.4
0.2	10	1.5
0.25	12.5	1.6
0.3	15	3.5
0.35	17.5	3
0.4	20	7
0.45	22.5	10
0.5	25	11
0.55	27.5	13
0.6	30	15
0.65	32.5	17
0.7	35	20
0.75	37.5	21
0.8	40	22
0.85	42.5	24
0.9	45	27
0.95	47.5	29
1	50	30

Figure 5.4 shows the Comparison of Traditional Network Coding Scheme and BSP-NC-ALGO-7 based on “Average Number of Transmissions per Node vs. Forwarding Factor”.

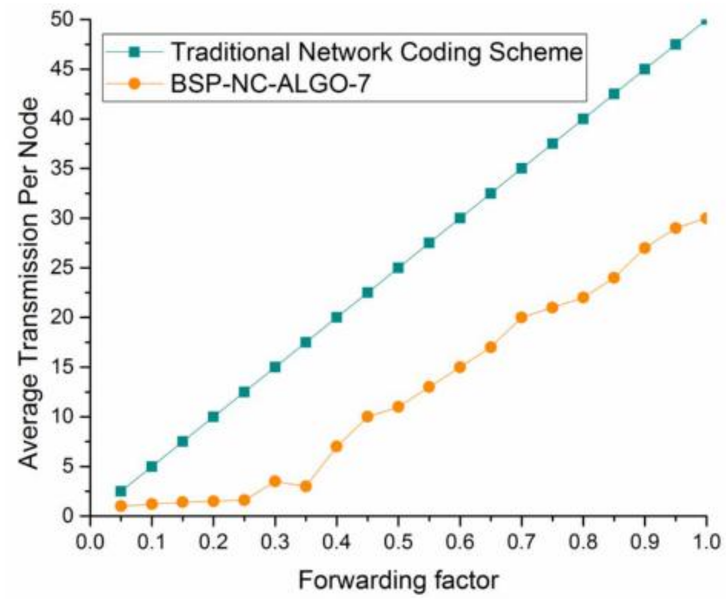

Figure 5.4 -Comparison of Traditional Network Coding Scheme and BSP-NC-ALGO-7 based on "Average Number of Transmissions per Node vs. Forwarding Factor"

Table 5.4 shows the number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-7.

Table 5.4: Number of Nodes vs. Number of Forwarding Nodes using BSP-NC-ALGO-7

Number of Nodes	Number. of Forwarding Node
20	8
30	10
40	15
50	17
60	22
70	29
80	30
90	35
100	37

Figure 5.5 shows the Number of Forwarding Nodes required for Network Coding vs. Number of Nodes using BSP-NC-ALGO-7.

Table 5.5 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-7).

Table 5.5: Comparison of BSP-NC-ALGO-(1-7)

Number of Nodes	Number of Forwarding Nodes using BSP-NC-ALGO-1	Number of Forwarding Nodes using BSP-NC-ALGO-2	Number of Forwarding Nodes using BSP-NC-ALGO-3	Number of Forwarding Nodes using BSP-NC-ALGO-4	Number of Forwarding Nodes using BSP-NC-ALGO-5	Number of Forwarding Nodes using BSP-NC-ALGO-6	Number of Forwarding Nodes using BSP-NC-ALGO-7
20	11	8	8	7	7	6	8
30	18	12	10	8	9	7	10
40	27	20	15	13	14	13	15
50	31	21	18	15	16	14	17
60	40	25	22	19	20	18	22
70	45	30	29	25	27	22	29
80	51	40	31	26	28	24	30
90	59	43	35	30	33	27	35
100	66	44	38	32	35	29	37

Figure 5.6 reveals the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-7).

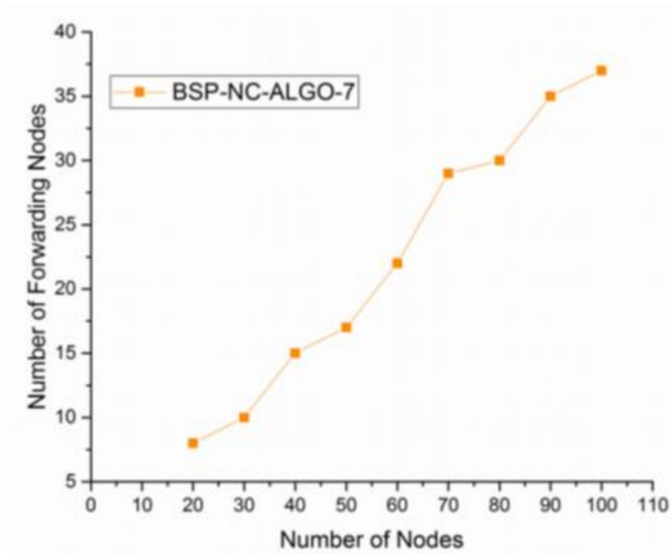

Figure 5.5: Number of Forwarding Nodes required for Network Coding vs. Number of Nodes using BSP-NC-ALGO-7

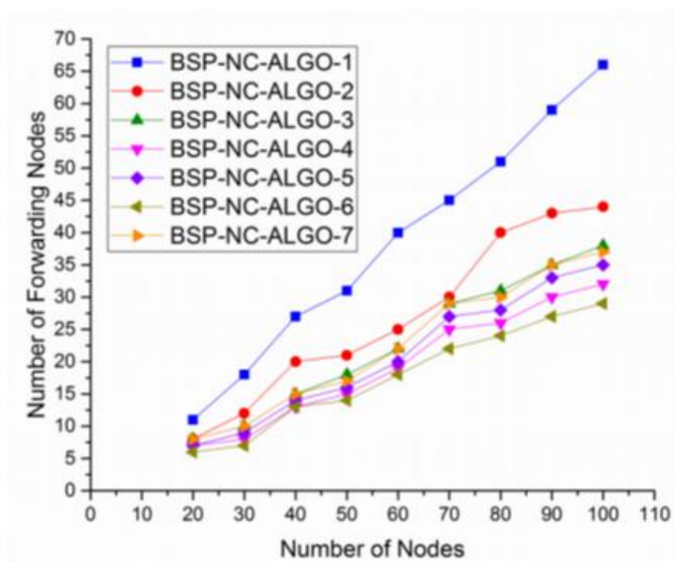

Figure 5.6: Comparison of BSP-NC-ALGO-(1-7)

Table 5.6 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-7) and previous broadcast storm problem subtracting schemes.

Table: 5.6: Comparison of Previous BSP subtracting schemes and BSP-NC-ALGO-(1-7)

Number of Nodes	Number of Forwarding Nodes -Probabilistic Scheme	SchemeNumber of Forwarding Nodes -Counter-based	SchemeNumber of Forwarding Nodes -Distance-based	SchemeNumber of Forwarding Nodes -Location-based	Number of Forwarding Nodes -Cluster-based Scheme	Number of Forwarding Nodes -BSP-NC-ALGO-1	Number of Forwarding Nodes -BSP-NC-ALGO-2	Number of Forwarding Nodes -BSP-NC-ALGO-3	Number of Forwarding Nodes -BSP-NC-ALGO-4	Number of Forwarding Nodes -BSP-NC-ALGO-5	Number of Forwarding Nodes -BSP-NC-ALGO-6	Number of Forwarding Nodes -BSP-NC-ALGO-7
20	16	15	14	13	12	11	8	8	7	7	6	8
30	24	22	20	20	19	18	12	10	8	9	7	10
40	32	31	29	29	28	27	20	15	13	14	13	15
50	40	37	35	33	32	31	21	18	15	16	14	17
60	48	45	42	42	42	40	25	22	19	20	18	22
70	56	52	48	47	46	45	30	29	25	27	22	29
80	64	58	55	54	53	51	40	31	26	28	24	30
90	72	67	63	62	61	59	43	35	30	33	27	35
100	80	72	68	67	67	66	44	38	32	35	29	37

Figure 5.7 shows the comparative analysis of number of forwarding nodes required for Network coding in comparison to actual number of nodes present in the system using BSP-NC-ALGO-(1-7) and previous broadcast storm problem subtracting schemes.

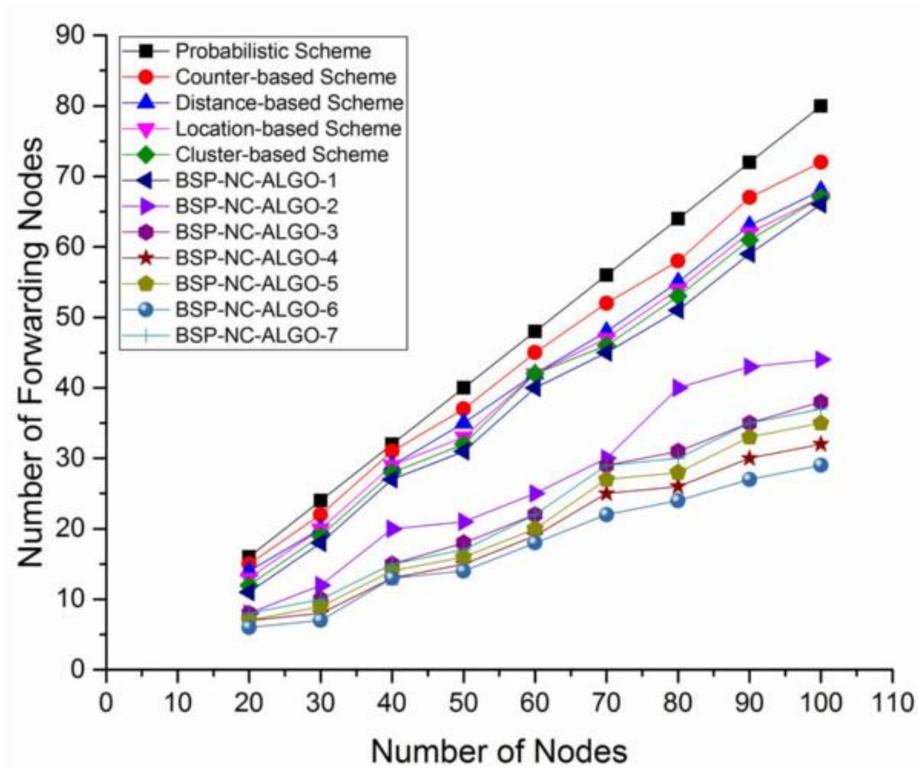

Figure 5.7: Comparison of Previous BSP subtracting schemes and BSP-NC-ALGO-(1-7)

5.4. Summary & Discussion

From above discussions, facts and figures, it is apparent that on integrating Network coding scheme with AODV protocol, algorithm formed is much more capable as compared to the elderly Network coding scheme as less number of average transmissions are needed per node. The improvement in the performance of the projected AODV integrated Network coding scheme upon the original Network coding scheme increases with increase in density. Therefore a basic modification of discovering a reduced sub-graph from first original sub-graphs utilizing AODV routing scheme can augment the execution of Network Coding to an awesome degree.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1. Conclusion

Regardless the fact that wireless is progressively the preferred mode of network access, the present wireless systems are not prepared to meet the demands of emerging high bandwidth applications. Current deployed wireless mesh work systems are fabricated using a structure established in wired network design, which at last limits throughput. This thesis advocates an elective architecture built around network coding and shows that it can give enormous cost saving by decreasing number of forwarding nodes that codes packet together.

Broadcasting in a MANET has generally extraordinary attributes from that in different networks. It could bring redundancy, collision and contention. Network coding may affect the outline of the design of new networking and information dissemination protocols. The simulation clearly demonstrates that considering neighborhood information when integrated with network coding architecture (BSP-NC-ALGO 1,2,3) requires less number of forwarding nodes in contrast with other algorithms like Counter-based scheme, Location-based scheme, Probabilistic scheme, Distance-based scheme and Cluster-based scheme to reduce the effects of Broadcast Storm problem .

Proposed Algorithm (BSP-NC-ALGO-4) utilizing the network attributes like incoming traffic degree, outgoing traffic degree, sum of input and output link bandwidth of a node of packet transmission requires less number of forwarding nodes in contrast with global network information based algorithm (BSP-NC-ALGO 1,2,3) and thus performs better. Besides, these algorithms utilize bit addresses to which brings down the measure of bytes sent per packet. Also, utilizing bit addresses (BSP-NC-ALGO-5) to help in recognizing the quantity of packets incorporated into each coded packet and permits to effortlessly distinguishing which packet are incorporated into the coded message.

SET Partition theory based algorithm Proposed Algorithm (BSP-NC-ALGO-6) when integrated with Network Coding approach distinguishes and comprises nodes based on the fact of bundles of different network coded reduces the Number of Forwarding nodes for retransmission in contrast to actual number of nodes and performs the best way in all proposed algorithms. The enhancement in the performance of the proposed AODV integrated Network coding scheme (BSP-NC-ALGO-7) over the original Network coding scheme increases with increase in density. Hence a basic alteration of discovering a reduced sub-graph from the first sub-graph utilizing AODV routing scheme can enhance the execution of Network Coding to an awesome degree.

This thesis incorporates the above thoughts into a practical network-coded wireless architecture. Further, it coordinates network coding inside the present stack, creates practical coding and decoding algorithms, gives model executions of the proposed structures, and testbed assessments that show huge execution gains in terms of number of forwarding nodes performing coding of packets together. Along these lines, we believe that this work puts forth a solid defense for an elective wireless network design dependent on network coding.

The frameworks in this thesis tended to the significant difficulties associated with carrying the hypothesis of network coding to practice and incorporating it into the present system stack. We have learnt innumerable noteworthy lessons while taking a shot at this usage. Especially, it pays to be opportunistic. The productivity of broadcasting gives off an impression of being straight forwardly related to the construction of a connected dominating set of minimal size.

6.2 Future Scope

At the end of the day, this dissertation is about building an inter-disciplinary approach to design future wireless networks. Future research includes applying these proposed algorithm to make network coded routing more scalable. A challenge in our view is how to control a lot of overhead where our proposed algorithm are promising and avoiding any overhead. We might attempt to ensure collision free transmission with nearly higher probability. The future work will be centered around the versatile procedure, permitting an expansion in the coding productivity in wireless systems. The new arrangement ought to effectively respond on connections quality, organize structure changes because of nodes versatility, and the dynamic idea of remote interchanges. It ought to likewise be free of the communicated transmission. At long last, it should consolidate some dimension of knowledge, considering us to modify the network coding procedures to a traffic type, steering data, and other conceivable cross layer information. In our future work, we need to check whether the gains related with network coding are likewise seen in scalable network devices.

REFERENCES

[1] C. Dandelski, B. Wenning, M. Kuhn, and D. Pesch, "Broadcast Storm Problem in Dense Wireless Lighting Control Networks," in *International Symposium on Wireless Communication Systems (ISWCS)*, pp. 91–95,2015.

[2] Y. Deng, Y. Yu, K. Liu, L. Li, and X. Su, "Broadcasting algorithm based on successful broadcasting ratio and energy balance of nodes in mobile ad hoc networks," *International Journal of Internet Protocol Technology*, vol. 11, no. 1, p. 1-11, 2018.

[3] Q. Yuan, R. Du, M. Zheng, Y. Xiang, J. Chen, and K. He, "Dominating Set and Network Coding-Based Routing in Wireless Mesh Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 2, pp. 423–433, 2014.

[4] K. M. Alzoubi, P. J. Wan, and O. Frieder, "New distributed algorithm for connected dominating set in wireless ad hoc networks," *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, vol. 2002-January, no. c, pp. 3849–3855, 2002.

[5] H. Lim and C. Kim, "Flooding in Wireless Ad Hoc Networks," *Computer Communications*, vol. 24, pp. 353–363, 2001.

[6] D. G. Reina, S. L. Toral, P. Jonhson, and F. Barrero, "Hybrid flooding scheme for mobile Ad hoc networks," *IEEE Communications Letters*, vol. 17, no. 3, pp. 592–595, 2013.

[7] V. Drabkin, R. Friedman, G. Kliot, and M. Segal, "On reliable dissemination in wireless Ad hoc networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 6, pp. 866–882, 2011.

[8] X. Wu, Y. Yang, J. Liu, Y. Wu, and F. Yi, "Position-aware counter-based broadcast for mobile Ad Hoc networks," *Fifth International Conference on Frontier of Computer Science and Technology*, pp. 366–369, 2010.

[9] A. Mohammed, M. Ould-Khaoua, L. M. Mackenzie, and J. Abdulai, "An adjusted counter-based broadcast scheme for mobile ad hoc networks," *Proc. - Tenth International Conference on Computer Modeling and Simulation*, pp. 441–446, 2008.

[10] J. C. Kuo and W. Liao, "Hop count distance in flooding-based mobile ad hoc networks with high node density," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 3, pp. 1357–1365, 2007.

[11] S. O. Al-Humoud, L. M. Mackenzie, M. Ould-Khaoua, and J. Abdulai, "Threshold analysis of adjusted counter-based broadcast in MANETs," *Proc. - Second UKSIM European Symposium on Computer Modeling and Simulation*, pp. 500–505, 2008.

[12] Li, Xiang-Yang, and Ivan Stojmenovic. "Broadcasting and Topology Control in Wireless Ad Hoc Networks." (2005): 239-261.

[13] J. Wu and W. Lou, "Forward-node-set-based broadcast in clustered mobile ad hoc networks," *Wireless Communications And Mobile Computing*, vol. 173, pp. 155–173, 2003.

[14] I. Stojmenovic and J. Wu, "Broadcasting and activity scheduling in ad hoc networks," *Mobile Ad Hoc Networking*, pp. 205–229, 2005.

[15] Ł. Apiecionek, "Limiting Energy Consumption by Decreasing Packets Retransmissions in 5G Network," *Mobile Information system*vol. 2017, pp. 1–9, 2017.

[16] D. K. Barai, S. Cherukuvada, G. Sridevi, R. Srikanth, and S. Umar, "Design and analysis of amplitude and divergence scheme for solving broadcast storm issues," *International Conference on Electrical, Electronics, and Optimization Techniques*, pp. 2116–2118, 2016.

[17] M. A. Hasnat, S. Dey, and A. Saha, "A novel technique of integrating the conventional algorithms for broadcasting protocols to reduce the redundancy in wireless ad-hoc networks 14th International Conference on Computer and Information Technology (ICCIT 2011), no. Iccit, pp. 664–669, 2011.

[18] K. Mahmood, T. Kunz, and A. Matrawy, "Adaptive Random Linear Network Coding with controlled forwarding for wireless broadcast," *2010 IFIP Wireless Days, WD*, 2010.

[19] S. Ghosh and L. Natarajan, "Linear Codes for Broadcasting with Noisy Side Information: Bounds and Code Constructions," *IEEE Inernational Symposium Information Theory - Proc.*, vol. 2018-June, no. c, pp. 441–445, 2018.

[20] A. Darehshoorzadeh, M. I. Sanchez, and A. Boukerche, "Modeling and analysis of opportunistic routing in multi-hop wireless networks," *Proc. - IEEE 22nd International Symposium on Modelling, Analysis & Simulation of Computer and Telecommunication Systems*, pp. 337–344, 2015.

[21] X. Zhao, C. T. Chou, J. Guo, S. Jha, and A. Misra, "Probabilistically reliable on-demand multicast in wireless mesh networks," *International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2008*, pp. 1–9, 2008.

[22] T. Camp, J. Boleng, B. Williams, L. Wilcox, and W. Navidi, "Performance comparison of two location based routing protocols for ad hoc networks," *Proceedings.Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 1678–1687, 2002.

[23] J. Wu, M. Cardei, F. Dai, and S. Yang, "Extended dominating set in ad hoc networks using cooperative communication," *Lecture Notes in Computer Science book series (LNCS, volume 3462)*, pp. 1393–1396, 2005.

[24] A. Kardan and M. Kajbaf, "Distributed broadcast minimum spanning tree (reliable version)," *Proc. 3rd International Conference on Intelligent Sensors, Sensor Networks and Information*, no. 2, pp. 681–686, 2007.

[25] C. H. Yen and H. C. Wang, "Minimum cost broadcast routing with forbidden set for Ad-Hoc networks," *Proc. WRI International Conference on Communications and Mobile Computing*, vol. 2, pp. 20–25, 2009.

[26] Jian Tang, Guoliang Xue, and Weiyi Zhang, "Power efficient broadcasting and multicasting in wireless networks with directional antennas," *IEEE International Conference on Communications*, pp. 1438–1442, 2005.

[27] B. Bellur and R. G. Ogier, "A reliable, efficient topology broadcast protocol for dynamic networks," *IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*, vol. 1, pp. 178–186, 1999.

[28] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "The energy efficiency of distributed algorithms for broadcasting in ad hoc networks," *The 5th International Symposium on Wireless Personal Multimedia Communications*, vol. 2, pp. 499–503, 2002.

[29] P. J. Wan, G. Călinescu, and C. W. Yi, "Minimum-power multicast routing in static ad hoc wireless networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 3, pp. 507–514, 2004.

[30] J. Iguchi-Cartigny, P. M. Ruiz, D. Simplot-Ryl, I. Stojmenovic, and C. M. Yago, "Localized minimum-energy broadcasting for wireless multihop networks with directional antennas," *IEEE Transactions on Computers* vol. 58, no. 1, pp. 120–131, 2009.

[31] V. Tiruchengode, "Scrutiny of broadcasting efficiency with and without clustering in Manet," *International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, pp. 101–105, 2017.

[32] E. Sivaraman, "Dynamic cluster broadcasting for Mobile Ad Hoc Networks," *International Conference on Communication and Computational Intelligence (INCOCCI)*, pp. 123–127, 2010.

[33] M. Singh, S. Lee, and T. Kit, “Cluster-based routing scheme for Wireless Mesh Networks,” *13th International Conference on Advanced Communication Technology (ICACT2011)*, pp. 335–338, 2011.

[34] A. Fathi and H. Taheri, “Enhance topology control protocol(ECEC) to conserve energy based clustering in Wireless Ad Hoc Networks,” *3rd International Conference on Computer Science and Information Technology*, vol. 9, pp. 356–360, 2010.

[35] A. Eryilmaz, D. S. Lun, and B. T. Swapna, “Control of multi-hop communication networks for inter-session network coding,” *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 1092–1110, 2011.

[36] 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, 2009.

[37] F. T. Zohra and A. Rahman, “Mathematical analysis of self-pruning and a new dynamic probabilistic broadcast for MANETs,” *International Conference on Networking Systems and Security (NSysS)*, pp. 1–9, 2015.

[38] H. Dhillon and H. Q. Ngo, “CQMP: a mesh-based multicast routing protocol with consolidated query packets,” *IEEE Wireless Communications and Networking Conference, 2005* pp. 2168–2174, 2005.

[39] V. D. Park and M. S. Corson. Temporally-ordered routing algorithm (TORA) version 1: Functional specification. Internet Draft, 1997.

[40] W. Peng and X. Cheng Lu, “On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks,” *First Annual Workshop on Mobile and Ad Hoc Networking and Computing. MobiHOC (Cat. No.00EX444)*, pp. 129–130, 2000.

[41] C. Perkins and E. M. Royer, “Ad-hoc on-demand distance vector routing,” *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90–100, Feb 1999.

[42] A. Qayyum, L. Viennot, and A. Laouiti, “Multipoint relaying for flooding broadcast messages in mobile wireless networks,” *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, vol. 2002-January, no. c, pp. 3866–3875, 2002.

[43] E. M. R. and C. K. Toh., “A review of current routing protocols for adhoc mobile wireless networks,” *IEEE Personal Communications*, vol. 6(2), no. April, pp. 46–55, 1999.

[44] I. Stojmenovic, M. Seddigh, and J. Zunic, “Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 1, pp. 14–25, 2002.

[45] J. Wu and H. Li, “On calculating connected dominating sets for efficient routing in ad hoc wireless networks,” *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, pp. 7–14, august 1999

[46] B. Das, R. Sivakumar, and V. Bharghavan, “Routing in ad-hoc networks using a virtual backbone,” Proc. ICCCN’97, pp. 1–20, Sept. 1997.

[47] R. Sivakumar, B. Das, and V. Bharghavan, “An improved spine-based infrastructure for routing in ad hoc networks,” *IEEE Symposium on Computers and Communications (ISCC) 98*, 1998. Athens, Greece.

[48] K. M. Alzoubi, P. J. Wan, and O. Frieder, “New distributed algorithm for connected dominating set in wireless ad hoc networks,” *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pp. 3849–3855, 2002.

[49] W. Lou and J. Wu, “Toward broadcast reliability in mobile ad hoc networks with double coverage,” *IEEE Transactions on Mobile Computing*, vol. 6, no. 2, pp. 148–163, 2007.

[50] Fei Dai and Jie Wu, “Distributed dominant pruning in ad hoc networks,” *IEEE International Conference on Communications, 2003. ICC '03*, vol. 1, pp. 353–357, 2004.

[51] M. A. Spohn and J. J. Garcia-Luna-Aceves, “Enhanced dominant pruning applied to the route discovery process of on-demand routing protocols,” *Proceedings. 12th International Conference on Computer Communications and Networks (IEEE Cat. No.03EX712)*, pp. 497–502, 2003.

[52] A. Rahman, M. Endadul Hoque, F. Rahman, S. K. Kundu, and P. Gburzynski, "Enhanced partial dominant pruning (EPDP) based broadcasting in ad hoc wireless networks," *International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, vol. 4, no. 9, pp. 895–904, 2009.

[53] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, vol. 4, no. 1, pp. 7–14, 2004.

[54] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Transactions on Mobile Computing*, vol. 1, no. 2, pp. 111–122, 2002.

[55] Tseng et al, "Wireless Networks," volume 8, pp 153–167, 2002.

[56] Narasimha M, inventor; Motorola Mobility LLC, assignee, "Wireless ad hoc network assembly using network coding," *United States patent US 9*,942,934. 2018 Apr 10.

[57] Haas Z et al, "Gossip-based ad hoc routing," *IEEE/ACM Transactions on Networking*, vol 14, no. 3, 2006, 479–491.

[58] Katti S et al, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol 16, no. 3, 2008, 497 –510.

[59] Li L, et al. , "Network coding-based broadcast in mobile ad-hoc networks," *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, ,pp. 1739–1747, 2007.

[60] Keller, Lorenzo, Emre Atsan, Katerina Argyraki, and Christina Fragouli, "SenseCode: Network coding for reliable sensor networks." *ACM Transactions on Sensor Networks (TOSN)* 9, no. 2 (2013): 25.

[61] Gnawali, Omprakash, et al. "CTP: An efficient, robust, and reliable collection tree protocol for wireless sensor network,." *ACM Transactions on Sensor Networks (TOSN)* 10.1 (2013): 16.

[62] Hui, Jonathan W., and David Culler. "The dynamic behavior of a data dissemination protocol for network programming at scale," *In Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp. 81-94. ACM, 2004..

[63] Hou, I-H., Y-E. Tsai, Tarek F. Abdelzaher, and Indranil Gupta. "Adapcode: Adaptive network coding for code updates in wireless sensor networks." *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications* pp. 1517-1525, 2008.

[64] Wang, Dan, Qian Zhang, and Jiangchuan Liu. "Partial network coding: Theory and application for continuous sensor data collection." *14th IEEE International Workshop on Quality of Service*, pp. 93-101, 2006.

[65] W. Shin et al., "Efficient 2-D mesh election algorithms for OFDMA-based wireless mesh networks", 5th International Conference on Signal Processing and Communication Systems (ICSPCS), vol. 2, pp. 68-73, 2012

[66] Agbaria, A., Hugerat, M., & Friedman, R., :Efficient and reliable dissemination in mobile Ad Hoc networks by location extrapolation," *Journal of Computer Networks and Communications*, pp. 1–11, 2011.

[67] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Transactions on Mobile Computing*, vol. 1,no. 2, pp. 111–123, 2002.

[68] C. Fragouli, J. Y. Boudec, and J. Widmer, "Network coding: An instant primer," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, 2006.

[69] Z. Wang, Y. Chen, and C. Li, "CORMAN: A novel cooperative opportunistic routing scheme in mobile Ad Hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 2, pp. 289–296, Feb. 2012.

[70] Swamy VN, Rigge P, Ranade G, Sahai A, Nikolić B, "Network coding for high-reliability low-latency wireless control," *IEEE Wireless Communications and Networking Conference* pp. 138-144, 2016.

[71] Chieochan S, Hossain E, "Cooperative relaying in Wi-Fi networks with network coding," *IEEE Wireless Communications*, Vol 19(2), pp. 57-65, 2012.

[72] Tan SE, Yew HT, Arifianto MS, Saad I, Teo KT, "Queue management for network coding in ad hoc networks," *Third International Conference on Intelligent Systems Modelling and Simulation*, pp. 657-662, 2012.

[73] Paschos GS, Fragiadakis C, Georgiadis L, Tassiulas L, "Wireless network coding with partial overhearing information," *Proceedings IEEE INFOCOM* Apr 14 pp. 2337-2345, 2013.

[74] Wang S, Tan G, Liu Y, Jiang H, He T, "Coding opportunity aware backbone metrics for broadcast in wireless networks," *IEEE Transactions on Parallel and Distributed Systems* pp. 1999-2009, 2014

[75] Kafaie S, Ahmed MH, Chen Y, Dobre OA. Performance analysis of network coding with IEEE 802.11 DCF in multi-hop wireless networks. IEEE Transactions on Mobile Computing. 2018 May 1;17(5):1148-61.

[76] Hsu YP, Abedini N, Gautam N, Sprintson A, Shakkottai S, "Opportunities for network coding: To wait or not to wait," *IEEE/ACM Transactions on Networking*. Vol 11;23(6) pp. 1876-89, 2015.

[77] Yassein MB, Al-Dubai AY, "Inspired Counter Based Broadcasting for Dynamic Source Routing in Mobile Networks," *IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computin,g* pp. 1455-1459,2015

[78] Eugster P et al, "Epidemic information dissemination in distributed systems," Computer, volume 37, no. 5, pp. 60 –67, 2004.

[79] Traskov D et al, "Network coding for multiple Unicasts: An approach based on linear optimization," *IEEE International Symposium on Information Theory*, pp. 1757–62, 2014.

[80] Fragoouli C et al, "Network coding," *ACM SIGCOOM Computer Communication Review*, volume 36, no. 1, pp. 63 –68, 2006.

[81] Marina M et al, "On-demand multipath distance vector routing in ad hoc networks," *Intl. Conf. on Network Protocols*, pp. 14 –23, 2001.

[82] Perkins C, et al, "Ad hoc on-demand distance vector (AODV) routing". (No. RFC 3561), 2003.

[83] Călinescu G, Măndoiu II, Wan PJ, Zelikovsky AZ, "Selecting forwarding neighbors in wireless ad hoc networks,"*Mobile Networks and Applications*, vol 1;9(2):pp. 101-11, 2004.

[84] Bevan Das SR, Bharghavan V, "Routing in ad-hoc networks using a virtual backbone," *In Proceedings of the 6th International Conference on Computer Communications and Networks (IC3N'97)* pp. 1-20, 1997

[85] Wu J, Li H, "On calculating connected dominating set for efficient routing in ad hoc wireless networks" *In Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications* , pp. 7-14, 1999.

[86] Wu Y, Chou PA, Zhang Q, Jain K, Zhu W, Kung SY, "Network planning in wireless ad hoc networks: a cross-layer approach" *IEEE Journal on selected areas in communications.* Vol 23(1), pp. 136-50, 2005.

[87] Chachulski S, Jennings M, Katti S, Katabi D, "Trading structure for randomness in wireless opportunistic routing" *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 169-180, 2007.

[88] Sendonaris A, Erkip E, Aazhang B, "User cooperation diversity--part I: system description" *IEEE transactions on communications*, vol 20;51(11):pp. 1927-38, 2003.

[89] Peng C, Zhang Q, Zhao M, Yao Y, "On the performance analysis of network-coded cooperation in wireless networks" *In IEEE International Conference on Computer Communications*, pp. 1460-1468, 2007.

[90] Chen Y, Kishore S, Li J, "Wireless diversity through network coding" *In IEEE Wireless Communications and Networking Conference*, Vol. 3, pp. 1681-1686, 2006.

[91] Nosratinia A, Hunter TE, Hedayat A, "Cooperative communication in wireless networks" *IEEE communications Magazine.* Vol 42(10), pp. 74-80, 2004.

[92] Bao X, Li J, "Matching code-on-graph with network-on-graph: Adaptive network coding for wireless relay networks," *Proc. Allerton Conf. on Commun., Control and Computing IL.*

[93] Duffield NG, Horowitz J, Presti FL, Towsley D, "Multicast topology inference from measured end-to-end loss," *IEEE Transactions on Information Theory*. Vol 48(1), pp. 26-45, 2002.

[94] Cáceres R, Duffield NG, Horowitz J, Towsley DF, "Multicast-based inference of network-internal loss characteristics," *IEEE Transactions on Information theory*. vol 45(7) pp. 2462-80, 1999.

[95] Ng TE, Zhang H, "Predicting Internet network distance with coordinates-based approaches," *In Proceedings Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, v*ol. 1, pp. 170-179, 2002.

[96] Vardi Y, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the American statistical association*. vol 91(433), pp. 365-77,1996.

[97] Bestavros A, Byers JW, Harfoush KA, "Inference and labeling of metric-induced network topologies," *IEEE Transactions on Parallel and Distributed Systems*. vol 16(11), pp. 1053-65, 2005.

[98] Koetter R, Médard M, "An algebraic approach to network coding," *IEEE/ACM Transactions on Networking (TON)*, vol 11(5), pp. 782-95, 2003.

[99] Wu Y, Chou PA, Kung SY, "Minimum-energy multicast in mobile ad hoc networks using network coding," *IEEE Transactions on communication, vol* 53(11), pp. 1906-18, 2005.

[100] Dougherty R, Freiling C, Zeger K "Insufficiency of linear coding in network information flow," *IEEE transactions on information theory.* Vol 51(8), pp. 2745-59, 2005.

[101] Balakrishnan H, Padmanabhan VN, Seshan S, Katz RH., "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE transactions on networking*.vol 5(6), pp. 756-69, 2007.

[102] Jaggi S, Sanders P, Chou PA, Effros M, Egner S, Jain K, Tolhuizen LM., "Polynomial time algorithms for multicast network code construction," *IEEE Transactions on Information Theory*, vol 51(6), pp. 1973-82, 2005.

[103] Jinying L, Yanping Y, Yan Y, Zhengqing Y, "Broadcasting algorithm based on successful broadcasting ratio of neighbor nodes in mobile ad hoc networks," *IEEE 17th International Conference on Communication Technology (ICCT)*, pp. 965-969, 2017

[104] Wang XY, Yu YP, Liu XX, Ning DS, Liu KY, Yu Y, "Distance and energy-balance based reliable broadcast in mobile ad hoc networks,". *International Conference on Wireless Communication and Sensor Network*, pp. 174-179, 2014.

[105] Adarbah HY, Ahmad S, Arafeh B, Duffy A, " Efficient broadcasting for route discovery in mobile ad-hoc networks," *IEEE International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, pp. 1-7, 2015.

[106] Patel PV, Kadhiwala B, "Broadcasting techniques for route discovery in mobile Adhoc network—A survey" *3rd IEEE International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 671-674, 2016.

[107] Ostovari P, Wu J, Khreishah, A., "Network coding techniques for wireless and sensor networks, "*In The art of wireless sensor networks*, pp. 129-162, 2014.

[108] Liu Y, Sung CW, "Quality-aware instantly decodable network coding," *IEEE Transactions on Wireless Communications.* VOL 13(3), PP. 1604-15, 2014.

[109] Bani Yassein M, Abed Al-Hameed B, Mavromoustakis C. , "Adaptive counter-based broadcasting scheme in mobile ad hoc networks,". *In Proceedings of the 1st ACM workshop on High performance mobile opportunistic systems*,pp. 47-52, 2012.

LIST OF PUBLICATIONS

Journal

[1] M. K. Goyal, S. P. Ghrera, and J. P. Gupta, "Improving the Average Number of Network Coded Transmission in AODV Routing Protocol with Network Coding Scheme," *Indian Journal of Science & Technology*, vol. 9, no. 32, pp. 1–9, 2016. **[SCOPUS]**

[2] M. K. Goyal, S. P. Ghrera, and J. P. Gupta, "Reducing the number of forward nodes from 1-hop nodes to cover 2-hop nodes with network coding," *Journal pf Telecommunication, electronic and Computer Engineering*, vol. 9, no. 3–6, pp. 13–17, 2017. **[SCOPUS]**

[3] M. K. Goyal, S. P. Ghrera, and J. P. Gupta, "Integration scheme of network coding and address bit vector in wireless network to acquire more reliability," *International Journal of Applied Engineering Research*, vol. 12, no. 22, pp. 12701–12706, 2017. **[SCOPUS]**

[4] M. K. Goyal, S. P. Ghrera, and J. P. Gupta, "Network Tomography Integrated Probe Tested Network Coding in Wireless Networks," *International Journal of Recent Technology and Engineering*, vol. 7, no. 5, pp. 166–171, 2019. **[SCOPUS]**

Conference

[5] M. K. Goyal, S. P. Ghrera, and J. P. Gupta. "Heterogeneous Rate-Network Coding in Wireless Networks." *PhD Forum / 2nd International Conference on Smart IoT Systems: Innovations in Computing* 2019. **[SCOPUS]**

[6] M. K. Goyal, S. P. Ghrera, and J. P. Gupta,. "Global Network Information Based Counter Specific Network Coding." *PhD Forum / 2nd International Conference on Smart IoT Systems: Innovations in Computing* 2019. **[SCOPUS]**

[7] M. K. Goyal, S. P. Ghrera, and J. P. Gupta, and Jai Prakash Gupta. "Inference of Topology SET Integrated Probe Tested Network Coding in Wireless Networks." *PhD Forum / 2nd International Conference on Smart IoT Systems: Innovations in Computing* 2019. **[SCOPUS]**

Response to Reviewers

Comments from Reviewer -1:

1. **Relevance to the Objectives defined for the research:** Comments on the research work in accordance to objectives being achieved as per the aim of the research work.

   The thesis report studied the broadcast storm problem, a propagation cost criterion was defined from the number of forwarding nodes. Also sought to link theory to practice by addressing the common aspects problems encountered in integrating network encoding into the current network stack.
   The proposed algorithm performs better. Than the objectives are relevant and the goals are achieved.

   Response: I appreciate the reviewer comment and thank for acknowledging the better performance of the proposed algorithm.

2. **Novelty/Originality:** Rate Novelty/Originality of the research work, comments about original contribution by the Scholar.

   The research presents a high degree of novelty and originality, which was clearly evidenced from the publications in newspapers (04) and conferences (03) from results obtained.

   Response: I appreciate and thank the reviewer for acknowledging the novelty of the research work.

3. **Technical correctness:** Comments on the research work in terms of Technical correctness.
   Organization and clarity: Comments on the presentation of the thesis.

   In my view, the text of the thesis was well organized and written in a clear and objective manner. However, I suggest that along with the title of the figures and tables, the source from which the information was collected should be highlighted. If

the picture or table was created by the author, I suggest that you put "Font: author" if it is an adaptation of another article picture "Font: adapted from [XX]".

Response: Now, Source is provided along with the figures wherever necessary in chapter 1 & 2.

4. **Comments to the Author to improve upon the research work:** Please provide detailed comments that will help to provide feedback to the authors.

   As evidenced in the section of future work, many tests have yet to be conducted to solidify doctoral research to expand their productivity in wireless networks, changes in structure, verification of gains in NC and flexibility.

   Response: I thank the reviewer again to point out future directions of the similar work with different challenges.

Comments from Reviewer -2:

The topic "Reduction of Forwarding Nodes to reduce Effect of Broadcast Storm Problem using Network Coding" undertaken for research is relevant to the design and deployment of Mobile Adhoc Networks (MANETs). MANETs are networks that may be required in many emergency situations where network deploying infrastructure is not feasible. Essentially, the communication medium is wireless and thus method of communication is broadcast which many a times suffers from broadcast storm resulting in collisions, congestion and redundancy. Proposed research targets to reduce the broadcast storm by reducing the number of forwarding nodes.

The proposed schemes present some work which is unique and an attempt has been made to improve the performance of MANETs by reducing broadcast messages. Schemes exploit use of neighbor information and degree of incoming/ outgoing traffic to minimize the forwarding nodes and consequently reduce the broadcast messages. The schemes proposed are simulated and compared across different parameter settings as well as with few existing approaches. Network simulator NS2 s used which is quite standardized platform and results abstained can be relied upon.

Thesis is well organized in chapters and follows standard format.

Response: I thank the reviewer for highlighting the positives. I also thank the reviewer for the constructive criticism expressed in the form of the points below. I have tried my best to address the concerns raised.

However, Following are few observations which needs clarification and could have further improved the research work:

1. **Thesis has numerous silly grammatical errors. Although contribution is good, but language of the thesis including abstract and conclusion is poor.**

   Response: I thank the reviewer for pointing out grammatical errors left in the thesis.

   a) I have tried my best again to rectify the typographic and grammatical errors by doing a thorough proof read of the thesis. Some sentences have been deleted and others have been paraphrased.
   b) Abstract and conclusion have been little rewritten to signify the vital points of the thesis of the research work performed. Summary is also rewritten at the end of the chapter to point out the outcomes of each proposed algorithm.

2. **No standard font and format is followed. Fonts of texts in diagrams and labels of tables/diagrams are too large.**

   Response:

   a) Labels of tables/diagrams are now reduced to font size 10.
   b) Times New Roman is used throughout the thesis.

3. **References do not follow IEEE formats. Different formats are used at different references.**

   Response: The references have been formatted in the desired IEEE format.

4. **Throughout the thesis the word "No. of" is used in place of "number of" which annoying and immature.**

   Response: The word "No. of" is replaced now with the word "number of" throughout the thesis.

5. **Problem statement could have been improved. What does the word "encountering" mean in every objective of problem statement?**

   Response:

   a) Problem statement has been little rewritten to improve the context for better understanding of the objective of the research work performed.
   b) The word "encountering" is now removed from problem statement paragraph for better understanding of the problem statement scenarios and proposed algorithms.

6. **Why sufficient comparison of proposed schemes with other existing standard schemes is not done? For example, BSP-NC-ALGO is compared only with itself on different parameter settings and same is the case with other proposed algorithms. For validation of results, proposed schemes must be compared and analyzed with standard similar existing approaches.**

   Response:

   a) Other existing standard schemes like Counter-based scheme, Location-based scheme, Probabilistic scheme, Distance-based scheme and Cluster-based scheme have been implemented & results are shown using table 3.7 & Figure 3.5 in chapter 3.

b) Comparison study of our proposed algorithm with other existing standard schemes is shown using table 5.6 and Figure 5.7 in chapter 5.
c) BSP-NC-ALGO (1, 2 & 3) incorporates probability of transmission of packets, thus it makes the comparison valid in contrast to Probabilistic scheme.
d) The other proposed algorithms have considered different factors like incoming & outgoing traffic degree, address bit vector, SET partition theory and AODV integration with network coding which are not yet taken into consideration for solution of broadcast storm problem and thus makes this as an emerging area for research.